Índice

Índice

Índice

Índice

Índice

1 - Introdução

O Linux é um clone UNIX de distribuição livre para PCs baseados em processadores 386/486/Pentium.

O Linux é uma implementação independente da especificação POSIX, com a qual todas as versões do UNIX padrão (true UNIX) estão convencionadas.

O Linux foi primeiramente desenvolvido para PCs baseados em 386/486/Pentium, mas atualmente também roda em computadores Alpha da DEC, Sparcs da SUN, máquinas M68000 (semelhantes a Atari e Amiga), MIPS e PowerPCs.

O Linux foi escrito inteiramente do nada, não há código proprietário em seu interior.

O Linux está disponível na forma de código objeto, bem como em código fonte.

O Linux pode ser livremente distribuído nos termos da GNU General Public License (veja apêndice).

O Linux possui todos as características que você pode esperar de um UNIX moderno, incluindo:

- Multitarefa real
- Memória virtual
- Biblioteca compartilhada
- "Demand loading"
- Gerenciamento de memória próprio
- Executáveis "copy-on-write" compartilhados
- Rede TCP/IP (incluindo SLIP/PPP/ISDN)
- X Windows

A maioria dos programas rodando em Linux são freeware genéricos para UNIX, muitos provenientes do projeto GNU.

Muitas pessoas tem executado benchmarks em sistemas Linux rodando em 80486, e tem achado o Linux comparável com workstations médias da Sun e da Digital.

O Linux está disponível através da Internet por meio de centenas de sites FTP.

O Linux está sendo usado hoje em dia por centenas e centenas de pessoas pelo mundo. Está sendo usado para desenvolvimento de softwares, networking (intra-office e Internet), e como plataforma de usuário final. O Linux tem se tornado uma alternativa efetiva de custo em relação aos caros sistemas UNIX existentes.

Um exemplo de pacote de distrribuição do Linux mais populares é distribuido pela InfoMagic (http://www.infomagic.com, e-mail info@infomagic.com), a versão LINUX Developer’s Resource CD-ROM, de dezembro de 1996, contém 6 CD-ROMs, seu conteúdo sucinto é :

- Versão Red Hat 4.0 (instalando kernel 2.0.18)
- Versão Slackware 3.1 (Slackware 96 - instalando kernel 2.0)
- Versão Debian GNU/Linux 1.2
- X-Windows - Xfree86 version 3.2
- Arquivos Linux de tsx-11.mit.edu e sunsite.unc.edu
- Arquivos GNU de prep.ai.mit.edu

- Documnetação completa on-line & HOWTO’s (Guia de Instalação e Guia do Administrador da Rede, em inglês)
- Softwares demostração comerciais como : BRU, dbMan, StarOffice, Cockpit, Flagship, Smartware, GP Modula-2, Pathfinder, Scriptum, etc.

2. Historia do Linux

O Kernel do Linux foi, originalmente, escrito por Linus Torvalds do Departamento de Ciência da Computação da Universidades de Helsinki, Finlândia, com a ajuda de vários programadores voluntários através da Internet.

Linus Torvalds iniciou cortando (hacking) o kernel como um projeto particular, inspirado em seu interesse no Minix, um pequeno sistema UNIX desenvolvido por Andy Tannenbaum. Ele se limitou a criar, em suas próprias palavras, "um Minix melhor que o Minix" ("a better Minix than Minix"). E depois de algum tempo de trabalho em seu projeto, sozinho, ele enviou a seguinte mensagem para comp.os.minix:

> *Você suspira por melhores dias do Minix-1.1, quando homens serão homens e escreverão seus próprios "device drivers" ? Você está sem um bom projeto e esta morrendo por colocar as mãos em um S.O. no qual você possa modificar de acordo com suas necessidades ? Você está achando frustrante quando tudo trabalha em Minix ? Chega de atravessar noites para obter programas que trabalhem correto ? Então esta mensagem pode ser exatamente para você.*
>
> *Como eu mencionei a um mês atrás, estou trabalhando em uma versão independente de um S.O. similar ao Minix para computadores AT-386. Ele está, finalmente, próximo do estágio em que poderá ser utilizado (embora possa não ser o que você esteja esperando), e eu estou disposto a colocar os fontes para ampla distribuição. Ele está na versão 0.02... contudo eu tive sucesso rodando bash, gcc, gnu-make, gnu-sed, compressão, etc. nele.*

No dia 5 de outubro de 1991 Linus Torvalds anunciou a primeira versão "oficial" do Linux, versão 0.02. Desde então muitos programadores têm respondido ao seu chamado, e têm ajudado a fazer do Linux o Sistema Operacional que é hoje.

Ultimas versões do kernel do Linux

Release v1.0
    1.0.9
    Data: Sat Apr 16 21:18:02 UTC 1994

Release v1.1
    1.1.95
    Data: Thu Mar 2 07:47:10 UTC 1995

Release v1.2
    1.2.13
    Data: Wed Aug 2 12:54:12 UTC 1995

Release v1.3
    pre2.0.14
    Data: Thu Jun 6 19:30:56 UTC 1996

Release v2.0

2.0.28

Data: Tue Jan 14 12:33:26 UTC 1997

ftp://ftp.cs.Helsinki.FI/pub/Software/Linux/Kernel/v2.0/linux-2.0.28.tar.gz

Release v2.1

2.1.23

Data: Sun Jan 26 14:12:18 UTC 1997

ftp://ftp.cs.Helsinki.FI/pub/Software/Linux/Kernel/v2.1/linux-2.1.23.tar.gz

3 - Gerência de Processos

3.1 - Considerações Iniciais

Para explicarmos como o Linux gerência processos, faremos considerações iniciais sobre o código fonte do kernel do Linux (onde encontramos a implementação da Gerência de Processos) e a inicialização "boot" do sistema.

Neste tópico tentaremos explicar, de uma maneira ordenada o código fonte do Linux, tentando conseguir um bom entendimento sobre como o código fonte está situado e como as características mais relevantes do UNIX foram implementadas. O objetivo é ajuda-lo a se familiarizar com o projeto geral do Linux. Então, vamos começar por onde o Linux começa: seu sistema de boot.

Um bom entendimento da linguagem C é necessário para entender este material, assim como familiaridade com conceitos de UNIX e arquitetura dos PCs. Porém, nenhum código C aparecerá neste material, mas referencias de onde podem ser encontrados.

Qualquer referencia "pathname" à arquivos tem como ponto de partida a arvore principal de fontes, usualmente /usr/src/linux.

A maioria das informações reportadas aqui tem como referencia o código fonte do Linux versão 1.0. Referencias a versões posteriores conterão o símbolo novo.

Caso o símbolo não estiver presente, significa que não houveram modificações após as versões 1.0.9-1.1.76.

mais Ocasionalmente um parágrafo como este ocorrerá no texto. Indicando onde poderam ser obtidas mais informações sobre o assunto corrente (geralmente o código fonte).

3.1.1 - Inicialização ("boot" do sistema)

Quando o PC é ligado, o processador 80x86 encontra-se em modo real e executa o código contido no endereço 0xFFFF0, que corresponde a um endereço ROM-BIOS. O BIOS do PC realiza alguns testes no sistema e inicializa o vetor de interrupções no endereço físico 0. Depois disto ele carrega o primeiro setor do device bootavel em 0x7C00, e passa a execução para este endereço. O device é, usualmente, o disquete ou o disco rígido. A descrição anterior é um tanto simplificada, mas é tudo que se necessita para entender o trabalho inicial do kernel.

A primeiríssima parte do kernel Linux está escrito em linguagem assembly 8086 (boot/bootsect.S). Quando é executado, ele se move para o endereço absoluto 0x90000, carrega os próximos 2 kBytes de código do device de boot até o endereço 0x90200, e o resto do kernel para o endereço 0x10000. A mensagem "Loading..." é apresentada durante o carregamento do sistema. O controle é, então passado para o código contido em boot/Setup.S, outro código assembly de modo real.

A parte de "setup" identifica algumas características do sistema (hardware) e o tipo da placa VGA. Se requerido, pede ao usuário para escolher o modo do vídeo da console. E, então, move todo o sistema do endereço 0x10000 para o endereço 0x1000, passa para o modo protegido e passa o controle para o resto do sistema (endereço 0x1000).

O próximo passo é a descompressão do kernel. O código em 0x1000 vem de zBoot/head.S que inicializa os registradores e invoca decompress_kernel(), o qual é composto por zBoot/inflate.c, zBoot/unzip.c e zBoot/misc.c. O dado "descompresso"

vai para o endereço 0x100000 (1 Mega), e esta é a principal razão do por que o Linux não pode rodar com menos de 2 Megas de RAM.

mais O encapsulamento do kernel em um arquivo gzip é realizado por Makefile e utilitários no diretório zBoot. São arquivos interessantes para se dar uma olhada.

novo A versão 1.1.75 moveu os diretórios boot e zBoot para arch/i386/boot. Esta modificação pretendeu possibilitar a construção de "kernel verdadeiro" para diferentes arquiteturas.

O código "descompresso" é executado a partir do endereço 0x1010000 , onde todo o setup 32-bit esta lotado: IDT, GDT e LDT são carregados, o processador e o co-processador são identificados, a rotina start_kernel é invocada. Os arquivos fonte das operações acima estão em boot/head.S. Este, talvez, seja o código mais difícil em todo o kernel do Linux.

Note que se algum erro ocorrer durante alguns dos passos precedentes, o computador irá travar. O sistema operacional não pode manipular erros enquanto não estiver totalmente operante.

start_kernel() reside em init/main.c. Tode de agora em diante esta codificado em linguagem C, exceto gerência de interrupções e chamadas de sistemas (Bem, a maior parte das macros possuem códigos assembly embutidos, também).

Depois dos procedimentos com todas as questões iniciais, start_kernel() inicializa todas as partes do kernel, especificamente:

- Inicializa a memória e chama paging_init().
- Inicializa os traps, canais IRQ e scheduling.
- Se requerido, aloja um profiling buffer.
- Inicializa todos device drives e buffers de discos, bem como outras partes menores.
- Regula o delay loop (calcula o numero "BogoMips").
- Checa se a interrupção 16 está trabalhando com o co-processador.

Finalmente, o kernel está pronto para move_to_user_mode(), em seguida fork (bifurca) o processo de inicialização, cujos códigos estão no mesmo arquivo fonte. E o processo número 0, também chamado idle task (tarefa preguiçosa), se mantém rodando em um loop infinito.

O processo de inicialização tenta executar /etc/init, ou /bin/init, ou /sbin/init.

Se nenhum deles tem sucesso, o código se desvia para "/bin/sh /etc/rc" e cria um root shell no primeiro terminal (console). Este código é remanescente do Linux 0.01, quando o S.O. era feito para um kernel stand-alone, e não havia processo de login.

Depois de exec() o programa de inicialização de um dos lugares padrão (deve haver um deles), o kernel não tem controle direto sobre o fluxo do programa. Sua função, de agora em diante, é prover processos através de chamadas ao sistema (system calls), assim como prover eventos para serviços assíncronos (como uma interrupção do hardware). A multitarefa está inicializada, e inicializará o gerenciamento de acesso a multiusuários, através do fork() e processos de login.

Estando o kernel carregado e provendo serviço, vamos prosseguir dando uma olhada nesses serviços ("system calls").

3.2 - Gerência de processo pelo kernel

Do ponto de vista do kernel, um processo é uma entrada na tabela de processos. Nada mais.

A tabela de processos, então, é uma das mais importantes estruturas de dados no sistema, conjuntamente com a tabela de gerenciamento de memória e o buffer cache. O item individual na tabela de processos é a estrutura task_struct, definida em include/linux/sched.h. Com a task_struct, tanto informações de baixo quanto de alto nível, são mantidas – variando da cópia de alguns registradores de hardware até o inode do diretório de trabalho para o processo.

A tabela de processos é tanto um array quanto uma lista duplamente ligada, como uma árvore. A implementação física é um array estático de ponteiros, cujo tamanho é NR_TASKS, uma constante definida em include/linux/tasks.h, e cada estrutura reside em uma pagina de memória reservada. A estrutura da lista está entre os ponteiros next_task e prev_task, a estrutura em arvore é um tanto complexa, e não será descrita aqui. Voce pode desejar mudar NR_TASKS do seu valor default (que é 128), mas esteja certo de que há dependências, e será necessário recompilar todos os arquivos fonte envolvidos.

Depois do boot, o kernel está sempre trabalhando em um dos processos, e a variável global "current", um ponteiro para um item da task_struct, é usado para guardar o processo que está rodando. A variável "current" só é mudada pelo scheduler, em kernel/sched.c. Quando, porém, todos os processos necessitarem estar looked, a macro for_each_task é usada. Isto é consideravelmente mais rápido que uma procura seqüencial no array.

Um processo está sempre rodando em ou em "modo usuário" ou em "modo kernel". O corpo principal de um programa de usuário é executado em modo usuário e chamadas a sistema são executados em modo kernel. A pilha usada pelos processos netes dois modos de execução são diferentes – um seguimento de pilha convencional é usado para o modo usuário, enquanto uma pilha de tamanho fixo (uma página, cujo processo é dono) é usada no modo kernel. A página de pilha para o modo kernel nunca é swapped out, porque ela pode estar disponível sempre que um system call é introduzido.

Chamadas a sistema (System calls), no kernel do Linux, são como funções da linguagem C, seu nome "oficial" esta prefixado por "sys_". Uma chamada a sistema de nome, por exemplo, burnout invoca a função de kernel sys_burnout().

mais O mecanismo de chamadas a sistema (System calls) está descrito no capítulo 3 do Linux Kernel Hackers' Guide (http://www.redhat.com:8080/HyperNews/get/khg.html). Uma olhada em for_each_task e SET_LINKS, em include/linux/sched.h pode ajudar a entender a lista e a estrutura de árvore da tabela de processos.

3.3 - Criando e destruindo processos

Um sistema UNIX cria um processo através da chamada a sistema fork(), e o seu término é executado por exit(). A implementação do Linux para eles reside em kernel/fork.c e kernel/exit.c.

Executar o "Forking" é fácil, fork.c é curto e de fácil leitura. Sua principal tarefa é suprir a estrutura de dados para o novo processo. Passos relevantes nesse processo são:

- Criar uma página livre para dar suporte à task_struct
- Encontrar um process slot livre (find_empty_process())
- Criar uma outra página livre para o kernel_stack_page
- Copiar a LTD do processo pai para o processo filho
- Duplicar o mmap (Memory map - memoria virtual) do processo pai

sys_fork() também gerencia descritores de arquivos e inodes.

novo A versão 1.0 do kernel possui algum vestígio de suporte ao "threading" (trabalho ou processo em paralelo), e a chamada a sistema fork() apresenta algumas alusões à ele.

A morte de um processo é difícil, porque o processo pai necessita ser notificado sobre qualquer filhos que existam (ou deixem de existir). Além disso, um processo pode ser morto (kill()) por outro processo (isto é um aspecto do UNIX). O arquivo exit.c é, portanto, a casa do sys_kill() e de variados aspectos de sys_wait(), em acréscimo à sys_exit().

O código pertencente à exit.c não é descrito aqui - ele não é tão interessante. Ele trabalha com uma quantidade de detalhes para manter o sistema em um estado consistente. O POSIX "standard", por conseguinte, é dependente de sinais (flags), e tinha que trabalhar com eles.

3.4 - Executando Processos

Depois de executar o fork(), duas copias do mesmo programa estão rodando. Uma delas usualmente executa - exec() - outro programa. A chamada a sistema exec() deve localizar a imagem binária do arquivo executável, carrega-lo e executa-lo. "Carrega-lo" não significa, necessáriamente, copiar na memória a imagem binária do arquivo, para que, assim, o Linux possa atender a demanda de programas a serem executados.

A implementação Linux do exec() suporta formatos binários diferentes. Isto é dotado através da estrutura linux_binfmt, a qual embute dois ponteiros para funções - um para carregar o executável e o outro para carregar a "library" associada, cada formato binário deve conter, portanto, o executável e sua "library".

O sistema UNIX prove, ao programador, seis formas para a função exec(). Quase todos podem ser implementados como uma "library" de funções, e o kernel do Linux implementa sys_execve() independentemente das providas pelo UNIX. Ele executa uma única tarefa: carregar o cabeçalho do executável, e tenta executa-lo. Se

os dois primeiros bytes são "#!", então a primeira linha é ignorada e um interpretador é invocado, caso contrário o formato binário, registrado, é executado seqüencialmente.

O formato nativo do Linux é suportado diretamente por fs/exec.c, e as funções relevantes são load_aout_binary e load_aout_library. Assim como para os binários a função de carregamento "a.out" é invocada, e a função mmap() (memory map - memória virtual) aloca espaço em disco (no caso da memória real estar cheia) para o processo, ou invoca read_exec(), caso haja espaço em memória. "The former way uses the Linux demand loading mechanism to fault-in program pages when they're accessed, while the latter way is used when memory mapping is not supported by the host filesystem (for example the "msdos" filesystem)".

novo A partir da versão 1.1 do kernel, o Linux embutiu um sistema de arquivos (filesystem) revisado do msdos, que suporta mmap() (memory map - memória virtual). Além disso a estrutura linux_binfmt é uma "lista ligada" e não um array, para permitir carregar um novo formato binário como um módulo do kernel. Finalmente a estrutura, por si mesma, foi estendida para acessar rotinas com o formato relativo à core-dump.

4 - Gerência de Memória

4.1 - Gerenciamento de Memória do Linux (LMM)

A execução do LMM (Linux Memory Manager) exige uma estratégia de paginação com uma copy-on-write confiando nas 386 páginas auxiliares. Um processo alcança suas tabelas de páginas de seu parent (durante um fork) com as entradas marcadas como read-only ou trocado. Então, se o processo tenta escrever para este espaço de memória e a página é uma copy on write page, isto é copiado e a página marcada read-write. Um exec ( ) resulta na leitura de uma página ou mais do executável. O processo então erra em qualquer outra página que precisar.

Cada processo tem uma tabela de página que significa que pode acessar 1 Kb de tabela de página indicando para 1 Kb de 4 Kb, páginas que é 4 Gb de mémoria. Um diretório de página do processo é iniciado durante um Fork por copy-page-tables. O processo inativo tem seu diretório de página inicializado durante a sequência de inicialização.

Cada processo usuário tem uma tabela descritória local que contém um código de segmento e um segmento de dados. Estes segmentos usuários extendem de 0 para 3 Gb (0 X c 0000000). Nos espaços usuários, endereços lineares e endereços lógicos são idênticos.

No 80386, endereços lineares vão de 0 Gb para 4 Gb. Um endereço linear indica uma posição particular de memória dentro deste espaço. Um endereço linear não é um endereço físico --- isto é um endereço virtual. Um endereço lógico consiste de um seletor e um offset. O seletor indica para um segmento e o offset diz que distância na seção o endereço é localizado.

O código Kernel e o segmento de dados são seções privilegiados definidos na tabela descritora global e extende de 3Gb para 4Gb. O Swapper - page - dir é organizado para que estes endereços lógicos e físicos sejam idênticos no espaço Kernel.

O espaço 3Gb acima aparece no process page directory como indicadores para tabelas de páginas Kernel. Este espaço é invisível para o processo no user mode, mas o modo privilegiado é acionado, por exemplo, para sustentar um sistema de ligação. O modo surpevisor é inserido dentro do contexto do processo atual então a tradução do endereço ocorre com respeito ao diretório de página do processo, mas usando segmentos Kernel. Isto é idêntico no mapeamento produzido com o uso de swapper - pg - dir e segmentos Kernel como ambos diretórios de páginas usa a mesma tabela de página neste espaço. Apenas task [0] (A tarefa inativa, ás vezes chamada de "tarefa trocadora" por razões históricas, mesmo assim isto não tem relação com trocas nos implementos Linux) usa o swapper - pg - dir diretamente.

- O segmento base do processo usuário = o X 00, page - dir particular, para o processo.
- O processo usuário faz um sistema de ligação : segment base = 0 X c 0000000 page - dir = mesmo usuário page dir.
- swapper - pg - dir contém um mapeamento para todas as páginas físicas de 0 X 0000000 para 0 X c 0000000 + and_mem, então as primeiras 768 entradas em swapper - pg - dir são 0's, e então há 4 ou mais que indicam na tabela de páginas Kernel.
- O user page directories têm as mesmas entradas como swapper - pg - dir dos 768 acima. As primeiras 768 entradas mapeam o espaço usuário.

A vantagem é que sempre que o endereço linear é acima de 0 X c 0000000 tudo usa a mesma tabela de páginas Kernel (Kernel page Tables).

O monte usuário permanece no topo do segmento de dados do usuário e desce. O Kernel Stack não é uma bonita estrutura ou segmento de dados que eu possa apontar com um "aqui é um Kernel Stack". Um Kernel Stack_frame (uma página) é associada com cada novo processo criado e é usado sempre que o Kernel opera dentro do contexto deste processo. Coisas ruins aconteceriam se Kernel Stack descesse abaixo de seu corrente stack frame. [Onde o Kernel Stack é guardado? Eu sei que há um para cada processo, mas onde isto é armazenado quando isto não está sendo usado?]

Páginas usuários podem ser roubados ou trocados - Um user page é um que é mapeado abaixo de 3 Gb em uma tabela de páginas usuários. Esta região não contém page directories ou page tables. Apenas páginas sujas são trocadas.

Menores alterações são necessárias em alguns lugares (testes para limites de memória vem para a mente) para prover suporte para definidos segmentos programados. [Há agora uma modificação - |c| + O sistema de ligação usado por dosane, Wine, Twin, and Wabi para criar segmentos arbitrários.]

4.2 - Memória Física

Aqui está um mapa de memória física antes que qualquer processo de usuário for executado. A coluna da esquerda mostra o endereço de partida do item e os números em negrito são aproximados.

A coluna do meio mostra os nomes dos itens. A grande coluna da direita mostra a rotina relevante ou o nome variável ou explicações para ingresso.

* Projeto - Inits que adquirem memória são (principais.c) profil - buffer, com, init, psaux, init, rd, , init, scsi.dev - init.

Note que toda memória não marcada como livre é reservada (mem-init). Páginas reservadas pertencem ao Kernel e nunca estão livres ou trocadas.

Uma visão de memória do user process.

O código de segmento e dados do segmento extendem todo o caminho de 0 X 00 para 3 Gb. Correntemente o page fault handler do wp_page confere para assegurar que um processo não escreve para seu código de espaço.

De qualquer modo, pegando o sinal segu, é possível escrever para o code space, causando ocorrência de um copy - on - write. O Handler do_no_page assegura que qualquer página nova que o processo adquira pertença ao executável, uma biblioteca dividida, ao stack, ou dentro do valor do brK.

Um usuário de processo pode reordenar seu valor brK chamando sbrK ( ). Isto é o que malloc ( ) faz quando precisa. O texto e a porção de dados são distribuídos em páginas separadas ao menos que alguém escolha o N opção composta. A biblioteca dividida carrega endereços são correntemente tornadas da imagem dividida por ele mesmo.

O endereço é entre 1.5 Gb e 3 Gb, exceto em casos especiais.

4.3 - Distribuição da memória do processo usuário

O Stack, shlibs e os dados são muito afastados um do outro para serem spanned por uma tabela de página. Todas KPT são divididas por todos processo e deste modo eles não estão na lista. Apenas páginas sujas são trocadas. Páginas limpas são roubadas e deste modo o processo pode tê-los de volta para o executável se for

desejado. A maioria das vezes apenas as páginas limpas são divididas. Uma página suja termina dividida sobre um fork até que parent ou child escolham para escrever isto de novo.

Administração dos dados da memória na tabela do processo.

Aqui está um sumário de algum dos dados mantidos na tabela do processo que é usado para administração da memória.

Limites do processo da memória.

Ulong - start_code - and_code - and_data - brk, atart - stock

Erro de contagem de página.

Tabela do descritor local.
Sturct desc - sturct ldt {32} é a mesa descritora local para tarefa. Números de páginas residentes. Swappable - trocáveis Se então as páginas do processo não serão trocados. Kernel Stack page Indicador para a página distribuída no fork. Saved - Kernel - Stack V86 modo material (stuff) stract tss pilha de segmentos (stack segments) indicador da pilha Kernel Kernel stack pointer segmento da pilha Kernel Kernel stack segment (0X10) ssi = esp 2 = ss2 = 0

Níveis de previlégio não usados.

Segmentos seletores. Ds=es=fs=gs=ss=ok17,cs—

Todos indicam para segmentos no corrente 1 dt [ ]
c r 3 : indicam para o page directory para este processo
1 dt - LDT (n) seletores para tarefas correntes do LDT

4.4 - Inicialização da memória

No Start Kernel (main.c) há 3 variáveis relatadas para inicialização da memória:

memory_start	começa a 1 Mb atualizado pelo projeto de inicialização.
memory_end	término da memória física: 8 Mb, 16 Mb, ou qualquer outro.
Low memory_start	término do código Kernel e dados que é carregado inicialmente

Cada projeto init tipicamente torna memory_start e retorna um valor atualizado, se distribui espaços no memory_start (simplesmente pegando-a). Paging init ( ) inicializa a page-tables no { \ tt swapper - pg - dir} (começando a 0 X 0000000) para cobrir toda a memória física do memory_start para memory_end. Na verdade o primeiro 4 Mb é feito no startup_32 (heads).memory_start é incrementado se quaisquer nova page-tables são adicionados.

A primeira página é zerada para bloquear os indicadores das referências do alçapão nulo no Kernel.

No sched_init ( ) o 1 dt e tss descritores para tarefa [0] são postos no GDT, e carregado para dentro do TR e LDTR (a única vez que isto é feito explicitamente). Um trap gate (0X80) é ordenado para system-call.( ).

A bandeira tarefa aninhada é desligada na preparação para entrada do modo usuário: O cronômetro é ligado. O task-struct para task [0] aparece por inteiro em < linux / sched.h >

mem_map é então construído por mem_init ( ) para refletir o corrente uso das páginas físicas. Este é o estado refletido no mapa da memória física da seção anterior. Então Dinux move para dentro do modo usuário com um iret após empurrar o corrente ss, esp, etc.

Claro que o segmento usuário para task [0] são mapeados bem sobre os segmentos Kernel e deste modo a execução continua exatamente onde isto termina.

Task [0]:

pg_dir = swapper - pg - dir que sigmifica apenas endereços mapeados estão no alcance 3 Gb para 3 Gb + High memory.

LTD [1] = código usuário, base = 0 x 0000000, tamanho = 640 K
LDT [2] = dados usuários, base = 0 x 0000000, tamanho = 640 k

O primeiro exec ( ) põe a LTD entrada para task [1] para os valores usuários da base = 0x0, limite = task_size = 0 x c 0000000. Depois disso, nenhum processo vê os segmentos Kernel enquanto no modo usuário.

Processos e a Administração da Memória.

Memória relacionada trabalho feito por fork ( ):

- distribuição de memória
- 1 página para o Task-struct
- 1 página para o Kernel Stack
- 1 para o pg_dir e algumas para pg_tables (cópias - páginas - tabelas)
- Outras mudanças
- sso põe para o segmento Kernel stack (0x10) para ter certeza?
- espo põe para o topo da nova distribuição Kernel - stack - page.
- c r 3 põe por copy - page - tables ( ) para indicar para nova página de diretório distribuída

- 1 dt = LDT (task_nr) cria novo 1 dt descritor
- descritores põe no gdt para novo tss e 1 dt [ ]
- Os registros restantes são herdados do parent.

Os processos resultam dividindo seus códigos e segmentos de dados (embora eles tenham tabelas descritoras locais separados, as entradas indicam para os mesmos segmentos). O stack e páginas de dados serão copiados quando o parent ou child escreve para eles (copy-on-write).

Memória relacionada trabalho feito por exec ( ):

- distribuição de memória
- 1 página para exec header para omagic
- 1 página ou mais para stack (max_arg_pages)
- clear-página-tables ( ) usado para remover páginas velhas.
- change 1 dt ( ) põe os descritores no novo 1 dt [ ]
- 1 dt [1] = código base = 0 x 00, limite = task - size
- 1 dt [2] = data base = 0 x 00, limite = task - size

Estes segmentos são dpl = 3, p=1, s=1, g=1. Tipo = a (código or 2 dados)

- Eleva para MAX_ARG_PAGES páginas sujas de arqu e enup são distribuídos e guardado ao topo do segmento de dados para o novo usuário pilha criado.
- Ponha os indicadores de instrução do caller cip = ex.a_cutry
- Ponha o stack indicador do caller para o stack criado (esp=stack indicador). Este serão eliminados do Stack quando o caller resume.

Limites de Memória Atualizados

- cud_code = ex.a_text
- cud_data = cud_code + &x.d_data
- brK = end_data + ex.ª_bss

Interrupções e traps são sustentadas dentro do contexto da corrente tarefa. Em particular, o diretório de páginas do corrente processo é usado na tradução de endereços. Os segmentos, de qualquer modo, são segmentos Kernel para que todos os endereços lineares apontem para dentro da memória Kernel quer acessar uma variável no endereço 0 x 01. O endereço linear é 0 x 00000001 (usando segmentos Kernel) e o endereço físico é 0 x 01. O último é porque a página do processo diretório mapea esta extensão exatamente como page_pg_dir.

O espaço Kernel (0 x c 0000000 + high - memory) e mapeado pela tabela de páginas Kernel que são eles mesmos parte da memória reservada. Eles são consequentemente divididas por todos processos. Durante um fork copy-page-tables ( ) trata tabela de páginas reservadas diferentemente. Isto põe indicadores no diretório de páginas de processo para indicar para tabelas de página Kernel e na verdade não distribui novas tabelas de páginas como isto faz normalmente. Como um exemplo o Kernel - Stack - page (que ocupa algum lugar no espaço Kernel) não precisa de um associado page - table distribuídos no pg-dir do processo para mapeá-lo.

O interruptor de instruções põe o indicador stack e o segmento stack do privilégio valor salvo no Tss do corrente task. Note que o Kernel stack é um objeto realmente fragmentado - Isto não é um objeto único, mas sim um grupo de stack frames. Cada um distribuído quando um processo é criado e deixado quando ele sai. O Kernel stack não deveria crescer tão rapidamente dentro de um contexto de um processo que extende abaixo da corrente frame.

4.5 - Adquirindo e liberando memórias

Quando qualquer rotina Kernel precisa de memória isto acaba chamando get-free-page ( ). Este está num nível mais baixo do que Kmallor ( ) (de fato Kmalloc ( ) get-free-page ( ) quando isto precisa mais memória).
Get-free-page ( ) toma um parâmetro, a prioridade.

Possíveis valores são gfp_buffer_gfp, Kernel, gfp,nfs e gfp atomic. Isto tira uma página do the free-page-list, atualizados mem_map, zeram a página e retorna o endereço físico da página (note que Kmalloc) retorna um endereço físico. A lógica do mm depende do mapa da identidade entre o endereço lógico e físico.

Isto é por ele mesmo bastante simples. O problema é claro, é que o free-page-list pode estar vazio. Se você não requisitar uma operação atômica, nesta etapa, você entra dentro do domínio de uma page stealing e que nós discutiremos em um momento. Como um último recurso (e para requisitos atômicos) uma página é separada do secundary-page-list (como você pode ter achado, quando páginas são libertadas, o secundary-page-list enche primeiro a manipulação atual da page-list e mem-map ocorre neste misterioso macro chamado remove-from-mem-queve ( ) que você provavelmente nunca quer investigar. O suficiente para dizer que interrupções são incapacitados. [Eu penso que isto deveria ser explicado aqui. Isto não é tão difícil...]

Agora de volta ao "Roubando páginas" get-free-page ( ) chame try-to-fre-page ( ) que chame repetidamente shrink_buffers ( ) e swap-out ( ) nesta ordem até conseguir liberar uma página. A prioridade é aumentada em cada iteration sucessiva para que estas duas rotinas processem suas page-sterling-loops mais frequentemente. Aqui está um exemplo do processo swap-out:

- Faça a tabela do processo e adquira uma swappable task, por exemplo, Q.
- Ache um user page-table (não reservado) no espaço de Q.
- Para cada página na tabela try-to-swap-out (page)
- Termina quando a página é liberada.

Note que swap-out ( ) (chamada try-to-free-page ( )) mantém variáveis estatísticas e deste modo isto pode resumir a procura onde terminar a chamada anterior try-to-swap-out ( ) examine os page-tables de todos usar process e obrigue o sterling policy:

1) Não brincar com as páginas (reserved) reservadas
2) Envelheçar a página se ela é marcada acessada (1 bit)
3) Não mexa com página adquirida recentemente (last-free-pages ( ))
4) Deixe páginas sujas com map-counts > 1 intocadas
5) Diminua o map-count das páginas limpas
6) Librere páginas limpas se elas não são mapeadas
7) Troque páginas sujas com um map-count de 1

De todas essas ações, 6 e 7 vão parar o processo poruqe eles resultam na liberação atual de uma página física.

A quinta ação resulta uma dos processos perdendo uma página limpa não dividida que não foi acessada recentemente (diminuindo Q à rss) que não é tão ruim, mas os efeitos cumulativos de algumas iterations pode atrasar o processo muito. No presente, há 6 iterations, deste modo uma página dividida por 6 processos pode ser roubada se está limpa. Page table então são atualizados e o TLB invalidado. O trabalho atual de liberar uma página é feito por free-page ( ), a complementação de get-free-page ( ). Isto ignora páginas reservadas, atualiza mem-map, e libera a página

e atualiza o page-list (s) se não é mapeada. Para troca (em 6 em cima), write-swap-page ( ) é chamada e não faz nada notável da perspectiva da administração da memória. Os detalhes de shink-buffers ( ) nos levaria muito longe. Essencialmente isto procura free "buffers" (buffers são uma parte da memória que segura informação temporariamente quando dados transferem de um lugar para outro) em seguida escreve buffers sujos, e depois começa com buffers ocupados e chama free-page ( ) quando pode liberar todos os buffers numa página.

Note que page directories, page-table, e reserved pages não são trocadas, roubadas ou envelhecidas. Eles são mapeadas no process page directories com reserved page tables. Eles são liberados somente na saída do processo.

The page Fault Handles

Quando um processo é criado por fork, ele começa com um page directoru e uma página ou mais do executável. Deste modo the page fault handles é a forte da maioria da memória do processo. The page fault handles do page-fault ( ) recupera o endereço faltando no registro c r 2. O código do erro (recobrado no sys-call.s) diferencia o acesso do user / supervisior e a região para o fault-write proteção de uma página faltando. O anterior é sustentado pelo do-wp-page ( ) e o posterior pelo do-no-page ( ). Se o endereço falatando é maior do que Task-Size, o processo recebe um SIGKILL [Por que este controle? Isto pode acontecer somente em Kernel mode por causa da proteção do nível do segmento. Estas rotinas tem algumas sutilezas como elas podem ser chamadas num interrompimento. Você não ode supor que é a tarefa corrente que está executando de-no-page ( ) sustenta três situações possíveis:

1) A página é trocada
2) A página pertence a biblioteca executável ou dividida.
3) A página está faltando – uma página de dados não foi distribuída

Em todas as causas get-empty-pgtable ( ) é chamada primeiro para assegurar a existência de uma page table que cobre o endereço falatando. No terceiro para providenciar uma página no endereço requerido e no caso de uma página trocada, swap-in ( ) é chamado. No segundo caso, o handles calls share-page ( ) para ver se a página pode ser dividida com algum outro processo. Se isto falhar leia a página do executável ou biblioteca (Isto repete a chamada para Share-page ( ) se um outro processo fez o mesmo enquanto isso). Qualquer porção da página fora do valor brK é zerada.

A página lida do disco é contada como um erro maior. Isto acontece com um swap-in ( ) ou quando é lida da executável ou uma biblioteca. Outras casos são consideradas erros menores (mim-flt). Quando uma página divisível é achada ela é corite-protected. Um processo que escreve para uma página dividida vai precisar passar por um do-wp-page ( ) que faz o copy-on-write.

Do-wp-page ( ) faça o seguinte:

- Mande SIGSEGV se qualquer usar process o está escrevendo para o corrente code-space.
- Se a página velha não é dividida, então simplesmente não proteja-o.

Senão get-free-page ( ) and copy-page ( ). A página adquirire a bandeira suja da página velha. Diminua a conta do mapa da página velha.

4.6 - Paginando (Paging)

Paginando é a troca numa base da página melhor do que os processos inteiros. Nós vamos usar trocando aqui para referir à "paginando" , uma vez que apenas Linux

página, e não trocar, e pessoas são mais acostumadas à palavra "Swap" / "trocar" do que "page" / "paginar". Kernel pages nunca são trocadas páginas limpas também não são escritas para trocar. Elas são liberadas e recarregadas quando é requerida. O trocador mantém um único bit de informação de envelhecimento nas Páginas acessadas bit da page table cutries - [O que são os detalhes de manutenção? Como isto é usado?]

Linux suporta múltiplos swap files ou projetos que podem ser ligados ou desligados pelas ligações de swapoff system. Cada swap file ou projeto é descrito por uma strut-swap-info.

O campo das bandeiras (SWP-USED ou SWP-WRITE ok) é usado para controlar acesso para o swap files. Quando SWP- WRITE ok é desligado, o espaço não vai ser distribuído neste arquivo. Isto é usado por Swapoff quando isto tenta de não usar um arquivo. Quando swapoff adiciona um arquivo de troca nova isto aplica SWP-USED. Um variável imóvel no Swap files armazena o número dos arquivos ativos correntemente ativos. Os campos lowest - bit e hihgest - bit limitam a região livre na pasta de troca e são usadas para adiantar a procura por espaço de troca livre.

O programa do usuário m | < swap inicializa um swap device ou file. A primeira página contém uma assinatura (swap-space) nos últimos 10 bytes, e contém um mapa de bit. Inicialmente 1's no bitmap significam páginas ruins A'1' no bitmap significa que a página correspondente é livre. Esta página nunca é distribuída deste modo a inicialização precisa ser feita somente uma vez.

The Syscall Swapor ( ) é chamado pelo user program swapon tipicamente de / etc / rc. Algumas páginas da memória são distribuídas por swap-map e swap-lockmap, swap-map contém um byte para cada página no swapfile. Isto é inicializado do bitmap para conter 0 para páginas disponíveis e 128 para páginas que não pode ser usadas. Isto é para manter uma conta das petições da troca em cada página no swap file. Swap-lockmap contém um bit para cada página que é usada para assegurar exclusão mútua quando lendo ou escrevendo swap-files.

Quando uma página da memória está para ser trocada, um índice para posição da troca é obtido com uma chamada para get-swap-page ( ). Este índice é deste modo guardado em bits 1-31 da page table entry para que a página trocada possa ser localizada pela page fault handles, do-no-page ( ) quando necessário.

Os 7 bits mais altos do índice dão o swap file (ou projeto) e os 24 bits mais baixos dão o número da página neste projeto. Isto faz até 128 swap files, cada um com espaço para mais ou menos 64 Gb, mas o espaço em cima devido o swap map seria grande. Ao invés o tamanho do swap file é limitado para 16 Mb, porque o swap map então toma 1 página.

A função swap-duplicate ( ) é usado por copy-page-tables ( ) para deixar o processo da child herdar páginas trocadas durante um fork. Isto somente incrementa a conta mantendo no Swap-map para aquela página. Cada processo vai trocar numa cópia da página separa quando acessá-la. Swap-free diminui a conta mantendo no swap-map. Quando a conta abaixa para 0 a página pode ser redistribuída por get-swap-page ( ). Isto é chamado cada vez que uma página trocada é lida na memória (swap-inc) ou quando uma página está para ser descartada (free-one-table ( ), etc).

4.7 - Gerenciamento de Memória Cache

4.7.1 - Arquitetura de Memória Cache do Linux (Linux Flush Architecture)

O TBL é mais uma entidade virtual do que um modelo estrito quanto a Linux flush architecture e concernida. As característica única são isto mantem em ordem o mapeamento do processo kernel de algum modo, queira softivare ou hardware.

Código específico de arquitetura pode precisar ser modificado quando o kernel tiver mudado um processo/mapeamento kernel.

O shell (um lugar seguro p/ guardar dinheiro ou coisas) esta entidade é essencialmente "memory state"/"estado da memoria" como o flush architecture o vê. Em geral isto tem as propiedades seguintes:

- Isto sempre vai segurar cópias de dados que podem ser visto como atualizado pelo processo local.

- O funcionamento próprio pode ser relacionado ao TLB e o mapeamento do processo/Kernel page de algum jeito, isto é para dizer que eles podem depender um do outro.

- Isto pode, numa configuração cached virtual, causar problemas "aliasing" se uma página fisica é mapeada no mesmo tempo da que duas páginas virtuais e por causa dos bits de um endereço usado para catalogar a linha cache, a mesma porção do dedo pode acabar residindo no cache duas vezes, deixando resultados incompativéis.

- Projetos e DMA podem ou não ter capacidade para ver a cópia de um dedo mais atualizado que resida no cache do processo local.

- Corretamente, é suposto que a coerência num ambiente multiprocessador é mantida pelo subsistema cache/memória. Isto que dizer que, quando um processador requerer um dado no memory bus de maneira e um outro processador tem uma cópia mais atualizada, de qualquer jeito o requesitor vai obter uma cópia atualizada que pertença um outro processador.

(NOTA: SMP arquiteturas sem hardware cache conferece mechaníms são realmente possíveis, o arquitetura current flush não sustenta isto corretamente, se em algum ponto o Zinux apontar em algum sistema onda isto é uma questão debatida, eu vou adicionar os ganchos necessários mas não vai ser bonito)

Sobre o que o Fluch Architecture se importa: sempre, a visão da administração de memória hardware de um conjunto de mapeamento do processo Kernel serão consistentes com aqueles do Kernel page tables.

Se o memory managemat kernel code faz uma modificação para a user process page modificando o dado via kernel space alias da página física subjacente, o fio controle de usuário vai ser o dado correto antes que é permitido continuar a execução, indiferente da cache architecture e/ou a semântica.

Em geral, quando o estado do espaço de endereço é mudado somente (em código genérico da administração da memória kernelnome de generic kernel management cade) o fluch architecture hook apropriado vai ser chamado descrevendo que o estado muda totalmente.

Sobre o que o flush architecture não importa: que o mapeamento do DMA "DMA/driver coerência. Isto inclui DMA mappings (no sentido do MMU mappings) e o cache/DMA dado consistência. Estes tipos des assuntos não devem esta no flush architecture, veja embaixo como eles devem ser manuseados.

Split Instrution/data cache consistência com respeitro as modificações feito

para processo de instrução de espaço realizado pelo código de sinal de despacho signal dispatch cade. De novo, veja embaixo como isto devem ser manuseado de um outro jeito.

As interfaces para a flushachitesture e como executá-los em geral todas as rotinas descritos embaixo vão ser chamados na sequência seguinte: Fluch-cache-foo(...);
modify-address-space ();
clush - tlb-foo (...)
a lógica aqui é: Isto pode ser ilegal num arquitetura dada por um pedaço de dado cache para ensitir quando o mapeamento por aquele dado não existe, portanto o flush deve ocorrer antes que a mudança é feita.

É possivél para uma arquitertura de MMU/TLB dada realizar um andamento da tabela hardware hardware table wolk dos kernel page tables, portanto o TLV flush é feito depois que os page tables terem sido mudados para que depois o hardware só pode carregar a cópia nova da informação de page table para o TLB
void flush - cache - all (void);
void flush - tlb - all (void);
Essas rotinas são para notificar o architecture specific cade que mapeamento do espaço do endereço kernel uma mudança foi feita ao kernel address space mappings, que significa que os mapeamentos de todos processos foram efetivamente mudados.

4.7.2 - Implementação da Memória Cache

Uma implementação deve:

- Eliminar todos os entradas do cache que são válidas neste momento quando flush-cache-all é invocado isto refere-se ao virtual cache architecture, se a cache is write-back, essa rotina vai submeter o dado da cache para memoria antes do que invalidar cada ingresso. Para caches físicos, não é necessário realizar uma ação já que mapeamento físico não tem ponto de apoio no address space translations.

- Para flush-tlb-all todos TLB mappings para o kernel address space devem ser feito consitente com os OS page tables de qualquer maneira. Norte que com um arquitetura que possua a nação

- Para flush-tlb-mm, o tlb/mmu hardware é para estar localizado num estado onde isto vai ver (agora corrente) kernal page table entradas para o espaço de endereço pelo mm-strust.

flush_cache_range(struct mm_struct *mm, unsigned long start,
unsigned long end);
flush_tlb_range(struct mm_struct *mm, unsigned long start,
unsigned long end);
uma chance para uma particular range do user address no adelrass space descrito pelo mm-struct passada esta ocorrendo. As duas notas acima para FLUSH - mm( ) relecianando a mm-struct passada aplicam-se aqui também.

- Para Flush-cache-range num virtualmente cached system, todas entradess cache que são nolidas pena a range partem para o fim no address space descrito pelo mm-struect são para ser invalidadas.

- Para Flush-tlb-range, qualquer ação necessária para causar o MMUITLB hardware não conter traduções estragados são para ser realizados. Isso significa que quaiquer traduções estão no Kernel page tables no range start para acabar no address space descrito pelo mm-struet são para que a administração da memoria hardware sera deste ponto avançado, por qualquer significado.

```
void flush_cache_page(struct vm_area_struct *vma, unsigned long address);
void flush_tlb_page(struct vm_area_struct *vma, unsigned long address);
```

Uma chance para uma única página no address dentro do usar space para o address space descrito pelo um area-struet passado esta ocorrendo. Uma efetivação, se necessária, pode obter na mm-struet associado para este address space via uma um -Flags. Este caminho em uma efetivação onde a instrução e dara space não são unificados, alguem pode conferir para ver se um-exee esta posto no uma-sum-flags para possivelmente avistar flushing o instruction space, por exemplos:
As duas notas acima para flush-*-mm( ) concermindo o mm-struct (passado indiretamente via uma -um-mm) aplica aqui também.

A implemetação deve também :

- Para flush-cache-range, num virtualmente cache systam, todas entradas cacha que são validas para a página no addrees no address space descrito pelo uma são para ser invalidados.

- Para flush-tlb-range, qualquer ação necessária para causar o MMU/TLB hardware para não conter traduções estragadas são para ser efetuadas. Isto significa que quaisquer traduções estão nos kernel page tables para a página no address space descrito pelo uma passado são para que a administração de memória hardware, serão vistas deste ponto avançado de qualquer maneira.

4.7.3 - Carregando o Flush-PAGE para a RAM (Unsigned Long Page);

Este é o patinho feio. Mas sera semântica é necessário em muitas arquiteturas que precisei para adicionar isto apra a arquitetura flush para linux. Brevemente, quando (como um exemplo) serve um kernel um enode cow, isto usa o “suposto” mapeamento de todas memorias fisicas no espaço kernal para efetuar a cópia da página em questão para uma nova página. Este apresenta um problema para caches virtualmente catalogados que são write-back escritos de volta na natureza. Neste caso, o Kernel toca duas páginas fisicas no espaço Kernel. A sequencia do código sendo descrito aqui essencialmente
parece como:

```
do_wp_page()
{
	[ ... ]
		copy_cow_page(old_page,new_page);
		flush_page_to_ram(old_page);
		flush_page_to_ram(new_page);
		flush_cache_page(vma, address);
		modify_address_space();
		free_page(old_page);
```

```
        flush_tlb_page(vma, address);
    [ ... ]
}
```

Alguns dos códigos atuais tem sido simplificados para propositos espesificos.

Considere um cache virtualmente catalogados que é escrito de volta write-back. Neste momento que a cópia da página acontece para o supisto espaço kernel, é possivel para usar space a visão da página original para estar no caches (no endereço do usuário, por exemplo, onde o erro esta ocorrendo). A cópia da página pode trazer este dado (para a página velha) dentro do caches. Será também colocado o dado (no novo suporte kernel mapeado da página) sendo copiado para dentro da cache, e para write-back escrever de volta chachas este dado vai ser sujo ou modificado no cache.

Em tal caso a memoria principal não será a cópia mais recente do dado. Os caches são estúpidos, então para a nova página que estamos dando ao usuário, sem forçar o dado cached no suposto kernel para a memória principal o processo será o conteúdo velho da página. (Por exemplo qualquer lixo que estarem lá antes da cópia ter sido feita pelo processamento COW acima).

4.7.3.1 - Exemplo concreto de flush-page

Considere um processo que divide uma página, lê somente READ-ONLY com maior uma tarefa (ou varias) no endereço virtual Ox2000, no usar space. E para propósito espesíficos deixe nos dizer que este endereço virtual mapeia para a página física 0x14000.

Se a tarefa 2 tenha escrever para a página lê apenas no endereço 0x2000 nós alteremos um esso e (eventual fragmento do código)
mente resultado no code fragment mostrando acima no do-WP-PAGE ( ).

O Kernel vai obter uma nova página para tarefa 2, deixe-nos dizer que esta e uma página física 0x2600, e deixe-nos tambem dizer que os mapeamentos do suposto Kernel para páginas físicas 0x14000 e 0x26000 podem residir em dias únicos linhas cache ao mesmo tempo buscando no esquema da linha catalogada deste cache.

O conteúdo da página e copiado do mapeamento Kernel para página física 0x14000 para uns para página física 0x26000.

Neste momento, numa arquitetura cache virtualmente catalogada write - back nos temos uma inconsistência potencial. O novo dado copiado dentro da página física 0x26000 não e necessário na memória principal neste momento, de fato isto poderá estar toda no cache apenas no suposto kernel do endereço físico.

Também, o (não modificando, por exemplo, limpo) dado para a (velha) página original esta no cache do suposto kernel para página física 0x14000, isto pode produzir uma inconsistência mais tarde, então para proteger isto e melhor eliminar as cópias cached deste dado também.

Deixe-nos dizer não escrevemos os dados de volta para a página no 0x256000 e nos apenas deixamos isto lá. Nos retornariamos para a tarefa 2 (Quem teve esta nova página agora mapeada no endereço virtual 0x2000) ele completaria sua escrita, então ele leria algumas outras porções de dados nesta nova página (por exemplo, esperando o conteúdo que existe lá antes). Neste momento seo dado e deixado no cache no suposto kernel para nova página física, o usuário obterá o que que estava na memória principal antes da cópia para sua leitura. Isto pode levar a resultados dasastrosos.

4.7.4 - Conteúdo de uma arquitetura virtual

Numa arquitetura cache virtualmente catalogada, fica o que foi necessário para fazer a memória principal consistente com a cópia cached da página passada do espaço kernel.

Nota: Isto é na verdade necessário para esta rotina invalidar linhos em um cache virtual que não escrito de volta é write - back na natureza. Para ver porque isto e realmente necessário, refaça o exemplo acima com a tarefa 1 e 2, mas agora fork ( ) ainda outra tarefa 3 antes dos erros do cow ocorreram, considere o conteúdo do caches no kernel e user space se a sequencia seguinte ocorre na exata sucessão:

1. Tarefa 1 lê uma parte da página no 0x2000
2. Tarefa 2 COW erra a página no 0x2000
3. Tarefa 2 efetiva suas escritas para a nova página no 0x2000
4. Tarefa 3 COW erra a página 0x2000

Mesmo em um cache não escrito devolta virtualmente catalogado, a tarefa 3 pode ver o dado incossistente depois do erro COW se FLUSH-PAGE-TO-RAM não invalida a página física do suposto kernel do cache.

VOID-UP-DATE

Embora não estritamente parte da arquitetura flush, em certas arquiteturas algumas operações e controles precisam ser eferuados aqui parea as coisas darem certo proporcionalmente e para o sistema manter-se consistente.

Em particular, para caches virtualmente catalogados esta rotina deve conferir para ver que o novo mapeamento que vem sendo adicionado pelo conente erro de página não adiciona um bad alias “para o user space”.

Um “Bad Alias” e definido como dois ou mais mapeamentos (pelo menos um dos quais e escrevivel) para duas ou mais o páginas que traduzem para a exata página física, e devido ao algarismo catalogado do cache pode também residir na única e mutualmente exclusiva linhas cache.

Se um BAD ALIAS é detectado, uma implementação precisa resolver esta inconsistência de alguma maneira, uma solução e andar através de todo os mapeamentos e mudar as page-tables para fazer estas páginas como não concreáveis se o hardware permite tal coisa.

As conferências para isto são muito simples, tudo que uma implementação precisa fazer é:

Se ((uma -Um - Flags 6 (Um - Write/Um - Shared)) confere sua potência mau supostas, então para o caso comum (mapeamento escrevíveis devidos são extremamente raros) apenas uma comparação é necessitada para sistemas COW CAHCES virtualmente catalogados.

4.7.5 - Implicações Referentes a Arquitetura

4.7.5.1 - Arquitetura baseada no Modelo SMP

Dependendo da arquitetura certos consertos podem ser necessários para permitir a arquitetura FLUSH para trabalhar num sistema SMP.

O principal assunto e se uma das operações FLUSH acima fazem que o sistema inteiro veja o FLUSH globalmente, ou o FLUSH e apenas garantido para ser

visto pelo processador local.

Em um último caso um CROSS CALLING MECHANISM é necessário. Os dois correntes sistemas SMP suportados no LiNUX (intel e space) usam inter-processor interrupts para “transmitir” a operação FLUSH e faz isto correr localmente em todo processador se necessário como um exemplo, no sistema SUNHM Space todos precessadores no sistema precisam executar o pedido FLUSH para garantir a consistência através do sistema inteiro. De qualquer modo, nas máquinas SUNHD Space, TLB FLUSHES efetivamente no processador local são transmitidos sobre o BUS-SYSTEM pelo hardware e desta forma uma ligação cruzada não e necessária

4.7.5.2 - Implicações para arquitetura baseados no contexto MMU/CACHE.

A idéia inteira por trás do conceito de MMU e facilidades do contexto cache é para permitir muitos ADDRESS SPACES para dividir os recursos CACHE/MMU no CPU.

Para levar total vantagem de tal facilidade, e ainda manter a coerência descrita acima, requer-se algumas considerações extras do implementador.

As questões envolvidas variam muito de uma implementação para outro, pelo menos esta tem sido a experiência do autor. Mas em particular algumas destas questões são provavelmente para ser:

- A relação do mapeamento do espaço Kernel para os USER-SPACE, num contexto são **convertidas**, alguns mapeamentos do sistema kernel tem um atributo global, naquele o hardware não **concerde** ele mesmo com o contexto da informação quando uma tradução é feita, que tem seu atributo. Desta forma um FLUSH (em qualquer contexto) de um mapeamento de um Kernel CACHE/MMU poderia ser suficiente.

De qualquer maneira e possível um outros implementações para o Kernel para dividir o contexto chave associado com um ADDRESS SPACE particular. Pode ser necessário em tal caso andar por todos contextos que são contentemente válidos e efetuam o Flush completo em cada um para um Kernall Address Space Flush.

O custo por contexto Flush podem tornar uma questão chave, especialmente com respeito ao TLB. Por exemplo, se um Tlb Flush e necessário, em um grande Range de endereços (ou um inteiro Address Space) pode ser mais prudente distribuir e assumir um nova contexto MMU/para este processo por causa da eficiência

4.7.6 - Como tratar o que a arquitetura flush não executa com exemplos

A arquitetura Flush descrita não faz emendas para coerência de projetos DMA com dados Cached. Isto também não tem provisões para nenhuma estratégia de mapeamento necessários pelo DMA e projetos se forem necessários em um certa máquina Linux é Portad To.

Nenhuma destas questões são para a arquitetura Flush.

Tais questões são negociadas mais claramente no nível do Driver do projeto. O autor está mais convencido disto depois de sua experiência com um conjunto comum de sparc device drivers que precisaram de toda função corretamente em mais do que uma hand full de cache/mmu e bus architetures no mesmo kernel. De fato esta implementação é mais eficiente porque o motorista sabe exatamente quando o DMA precisa ver o dado consistente ou quando o DMA está indo criar uma inconsistência

que deve ser resolvida. Nenhuma tentativa para atingir este nivel de eficiencia via cochetes soma ao codigo de administracao generica da memoria kernel seria complexo e muito obscura como um exemplo, considere no sparc como os DMA buffers são manuscrito. Quando um device driver deve efetuar o DMA para/de um único buffer, ou uma dispersa lista de muitos buffers, ele usa um conjunto de rotinas abstratas.

Char * (*mmu_get_scsi_one)(char de char *, unsigned linux_sbus longo de struct *sbus);
sem (*mmu_sglist (*mmu_get_scsi_sgl)(struct de efeito *, int, linux_sbus de struct *sbus);
sem (*mmu_release_scsi_one)(char de efeito *, unsigned linux_sbus longo de struct *sbus);
sem (*mmu_sglist (*mmu_release_scsi_sgl)(struct de efeito *, int, linux_sbus de struct *sbus);
sem (*mmu_map_dma_area)(unsigned de efeito addr longo, len de int);

Essencialmente o mmu_get_* rotinas são passadas por um indicador ou um conjunto de indicadores e especificações de tamanho para áres no espaço kernel para que o DMA ocorra, eles retornam para o endereço capaz do DMA (por exemplo um que pode ser carregado do controlador do DMA para o transferidor). Quando o driver é feiro como DMA e o transferidor tiver completado com o(s) endereço(s) DMA para que recursos possam ser liberados (se necessario) e cache flushes possam ser efetivados (se necessario). A rotina ter um bloqueio de memoria de DMA por um longo periodo de tempo, por exemplo, um motorista de networking usaria isto para uma transmissao de pesquisa ou receber buffers. O argumento final é uma entidade especifica Sparc que permite o codigo do nivel da maquina efetuar o mapeamento se o mapeamento do DMA são ordenados em uma base por-bus.

4.7.7 - Questões abertas na Arquitetura Cache

Há pareceres para muita estupidas arquiteturas cache lá fora que queira causar problemas quando um alias está situado dentro do cache (mesmo um protegido onde nenhuma das entradas do cache suposto são escreviveis!). Da nota está o mipsr4000 que dará uma exceção quando tal situação ocorre, elas podem ocorrer quando o processamento cow está acontecendo na corrente implementação. No mais chips que fazem algo estupido como isto, um exception handler pode flush as entradas no cache que está sendo reclamado e tudo está em ordem. O autor esta mais concernido sobre o custo dessas exceções durante o processamento cow e seus efeitos que ocorrerão na performance cow, que essencialmente está para flush um user space page e se não o fazendo então causaria os problemas acima descritos.

Tem sido tardiamente aquecida a conversa sobre muito inteligentes networking hardware. Pode ser necessario estender a arquitetura flush para prover as interfaces e facilidades necessarias para estas mudanças para o codigo networking. É claro que, a arquitetura flush é sempre sujeita a aperfeiçoamentos e mudanças para buscar novas questões ou novos hardwares que apresentam um problema que estava até este ponto desconhecido

5. Sistema de Arquivo no Linux (File System)

5.1. - Conceitos Fundamentais

5.1.1 - Arquivos

Conceitualmente, arquivos são mecanismos de abstração que fornecem uma forma de armazenar e recuperar informações em disco. A característica mais importante de qualquer mecanismo de abstração é a forma de identificar os objetos com os quais o mecanismo trata.

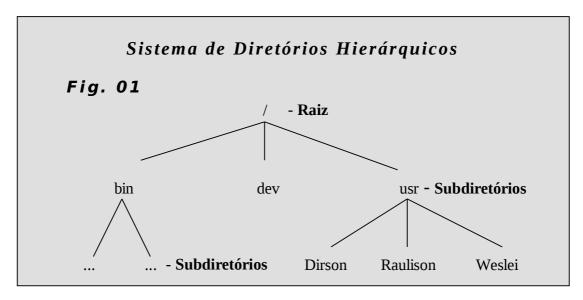
Quando um processo cria um arquivo, é preciso que tal arquivo receba um nome, normalmente dado pelo processo. Quando tal processo termina sua execução, o arquivo continua a existir, podendo ser acessado por outros processos, usando para tanto o nome atribuído ao arquivo.

O Linux faz distinção entre nome maiúsculos e minúsculos. Normalmente um nome de arquivo é composto de nome e uma extensão, separada por ponto. No Linux, o tamanho da extensão, se houver, fica a critério do usuário, e uma arquivo pode até ter duas ou mais extensões, exemplo : prog.c.Z.

Não há limite de números de caracteres utilizados para dar nome a arquivos.

O Sistema Operacional Linux, olha o arquivo como uma seqüência de byte, sem nenhuma estrutura, isto dá uma flexibilidade espantosa ao sistema de arquivo. Os programas de usuários, podem colocar o que desejarem nos arquivos e identificá-los da forma que lhe for mais conveniente, o Unix não influência em NADA nesta processo de identificação.

5.1.2 - Diretórios

Para tratar dos arquivos, o sistema operacional normalmente lança mão dos diretórios, no caso do Linux diretórios hierárquicos (vide figura 01). Os diretórios são um tipo de arquivo.

No Linux todos os arquivos fazem parte de um diretório, assim eles são mantidos e organizados, os diretórios são meios de oferecer endereços dos

arquivos, de maneira que o SO possa acessá-los rapidamente e facilmente. Ao entrar pela primeira vez em sua conta, o usuário já está em um subdiretório denominado subdiretório de entrada.

5.1.3 - Conta

É uma senha que é aberta pelo administrador do sistema (denominado de **root**) onde o usuário identifica-se para o computador, que então dá acesso ao seu diretório de entrada, onde você pode executar os comandos permitidos a sua senha. Nos SO padrão Unix, a conta é obrigatória para todos, a figura 02 mostra um exemplo de abertura de conta no Linux.

Figura 02

```
Linux 2.0.0 (carvalho.cpd.ufg.br) (tte p0)


carvalho login: root
Password:
Last login: Wed Jan 29 12:16:37 from
jacaranda.cpd.uf
Linux 2.0.0.
Carvalho:~$
```

5.1.4 - Tipos de arquivos

O Linux suporta arquivos regulares, arquivos de diretório, arquivos especiais de caracteres e arquivos especiais blocados.

Os arquivos regulares são aqueles que contém informações de usuários, por exemplos, tipo ASCII. Arquivos diretórios são arquivos usado na manutenção do sistema de arquivo. Arquivos especiais de caracteres estão diretamente ligado à entrada/saída e são usados para dispositivos seriais de entrada/saída, tais como terminais, impressoras e rede. Os arquivos especiais blocados são usado modelar dispositivos. Um exemplo de tipos de arquivos utilizados no Linux pode ser visto na figura 03.

Figura 03

```
carvalho:/usr$ ls
X11@            etc/             lib/         spool@
X11R6/          games/           local/       src/
X386@           i486-linux/      man/         tclX/
adm@            i486-linuxaout/  opemwin/     tkX/
bin/            i486-sesv4/      préerve@     tmp@
dict/           include/         sbin/
doc/            info/            share/
ftpusers        mtools.conf      sesog.conf
carvalho:/usr$
```

5.1.5 - Acesso a arquivos

O Sistema Operacional Linux, bem como os demais SO, trata o acesso a arquivos de forma radômica, ou seja, seus byte ou registros podem ser lidos em qualquer ordem.

5.1.6 - Atributos dos arquivos

Cada arquivo tem necessariamente um nome e um conjunto dados. Além disso, o Sistema Operacional associa a cada arquivo algumas outras informações que chamaremos de atributos de arquivos. A figura 04, nos mostra alguns dos atributos dos arquivos.

Figura 04

```
carvalho:/etc$ ls -l
total 11
lrwxrwxrwx   1 root   root        9   Dec   9 14:01 rmt -> /sbin/rmt*
-rw-r--r--   1 root   root      743   Jul  31  1994 rpc
-rw-r--r--   1 root   root       86   Jan  28  1994 securette
-rw-r--r--   1 root   root    21394   Dec   9 14:22 sendmail.000
-rw-r--r--   1 root   root    23580   Jan   6 12:28 sendmail.cf
drwxr-xr-x   2 root   root     1024   Dec   9 13:59 skel/
-rw-r--r--   1 root   root      314   Jan   9  1995 slip.hosts
-rw-r--r--   1 root   root      342   Jan   9  1995 slip.login
lrwxrwxrwx   1 root   root       13   Dec   9 13:59 utmp -> /var/og/utmp
lrwxrwxrwx   1 root   root       13   Dec   9 13:59 wtmp -> /var/og/wtmp
-rw-r--r--   1 root   root       76   Mae   8  1995 e p.conf.example
```

Como vimos neste exemplo, o Sistema de Arquivo do Linux permite restringir o acesso aos arquivos e diretórios permitindo que somente determinados usuários possam acessá-los. A cada arquivo e diretório é associado um conjunto de permissões. Essas permissòes determinam quais usuários podem ler, escrever, ou alterar um arquivo, e no caso de arquivos executáveis como programas, quais usuários podem executá-lo. Se um usuário tem permissão de execução de um diretório, significa que ele pode realizar buscas dentro daquele diretório, e não executá-lo como se fosse

programa. Passaremos a explicar a codificação, escolhemos aleatoriamente o sétimo arquivo **skel/** da figura 04 :

d	r	w	x	r	-	x	r	-	x	nome do arquivo
1	2	3	4	5	6	7	8	9	10	skel/

obs : o que está em negrito,caixa maior, corresponde a posição do arquivo skel/

- 1 - informa o tipo de arquivo (**d**⇨diretório,l ⇨ link, - ⇨demais arquivo)
- 2 - Permissões do Proprietário (**r** ⇨ leitura, , - não permitida leitura)
- 3 - Permissões do Proprietário (**w** ⇨ escrita, - não permitida escrita)
- 4 - Permissões do Proprietário (**x** ⇨ execução, - não permitida execução)
- 5 - Permissões do Grupo (**r** ⇨leitura, , - não permitida leitura)
- 6 - Permissões do Grupo (w ⇨ escrita, **-** não permitida escrita)
- 7 - Permissões do Grupo (**x** ⇨ execução, - não permitida execução)
- 8 - Permissões do Sistema (**r** ⇨ leitura, , - não permitida leitura)
- 9 - Permissões do Sistema (w ⇨ escrita, **-** não permitida escrita)
- 10 -Permissões do sistema (**x** ⇨ execução, - não permitida execução)

5.2 - Operações sobre arquivos

Os arquivos existem para armazenar informações e permitir a sua recuperação. As Chamadas de Sistemas mais comum relacionadas ao Sistema de Arquivo Linux são chamadas que operam sobre arquivos individuais ou envolvendo diretórios e sistema de arquivos como um todo .

A chamada **CREAT** não só cria um arquivo, mas também abre esta arquivo para escrita, indepedente do modo de proteção especificado para ele. O descritor de arquivo que a chama retorna, fd, pode ser usado para escrever no arquivo. Se a chamada CREAT for executada sobre um arquivo existente, esta arquivo será truncado para o comprimento 0, desde que os direitos do arquivos assim o permitam.

Para que um arquivo existente possa ser lido ou escrito, é necessário que ele seja primeiramente aberto e se ele esta aberto para leitura, escrita ou para ambas as operações. Várias opções podem ser especificadas. O descritor de arquivo que a chamada retorna pode então ser usado para leitura ou escrita. Posteriormente, o arquivo deve ser fechado através da chamada **CLOSE**, cuja execução torna o descritor de arquivo disponível para ser novamente utilizado numa chamada CREAT ou **OPEN** subseqüente.

A chamada **READ** é utilizada para ler o arquivo, os bytes lidos vêm em posição corrente de leitura. O processo que faz a chamada deve indicar a quantidade de informação a ser lida e providenciar um buffer para possibilitar a leitura.

A chamada **WRITE**, os dados são escritos no arquivo, geralmente a partir da posição corrente. Se tal posição for a de final de arquivo, o tamanho do mesmo cresce. Se a posição corrente no momento da escrita estiver no meio do arquivo, os dados existente nesta posição estaram perdidos para sempre, pois a operação de write escreve os novos dados em cima dos antigos.

Apesar da maioria dos programas ler e escrever arquivos sequëncialmente, em algumas aplicações os programas devem ser capaz de acessar randomicamente qualquer parte do arquivo. Associado a cada arquivo, existe um ponteiro que indica a posição corrente do arquivo. Quando a leitura ou escrita for seqüêncial, em geral, ele aponta para o próximo byte a ser lido ou a ser escrito. A chamada **LSEEK** têm três parâmetros: o primeiro do descritor de área para o arquivo, o segundo é a posição do

arquivo, o terceiro informa se a posição é relativa ao inicio do arquivo, à posição corrente ou final do arquivo. O valor que o **LSEEK** retorna é a posição absoluta no arquivo após a mudança no ponteiro.

Para cada arquivo o Linux mantem o modo do arquivo (regular, diretório ou arquivo especial), seu tamanho, o instante da última modificação, e outra informações pertinentes. Os programas podem verificar estas informações, usando a chamada **STAT**. Seu primeiro parâmetro é o nome do arquivo. O segundo é um ponteiro para a estrutura onde a informação solicitada deve ser colocada.

As chamadas do sistema relacionadas com os diretórios ou com o sistema de arquivo como um todo , em vez de um arquivo específicos. Os diretórios são criados utilizando as chamadas **MKDIR** e **RMDIR**, respectivamente. um diretórios o pode ser removido se estiver vazio.

A ligação de um arquivo cria uma nova entrada no diretório que aponta para um arquivo existente. A chamada **LINK** cria esta ligação. Os seus parâmetros especificam os nome originais e novo, respectivamente. As entrada do diretórios são removidas via **UNLINK**. Quando a última ligação para um arquivo é removida, é automaticamente apagada . Para um arquivo que nunca foi ligado, o primeiro **UNLINK** faz com que ele desapareça.

Os diretórios de trabalho é especificado pela chamada **CHDIR**. Sua execução faz com que haja mudança na interpretação dos nome dos caminhos realtivos.

A chamada **CHMODE** torne possível a mudança do modo um arquivo, ou seja, de seus bits de proteção.

5.3 - Arquivos Compartilhados

Quando vários usuários estão trabalhando juntos em um projeto, ele comumente precisam compartilhar arquivos. Em decorrência disto, muitas vezes é conveniente que um arquivo compartilhado apareça simultâneamente em diretórios diferentes que pertençam a diferentes usuários. A conecção entre um diretório e um arquivo compartilhado é chamada de ligação (link). O próprio sistema de arquivo é um gráfico acíclico dirigido , ou **DAG**, em vez de árvore. No Linux os blocos do disco não são listados no diretório, mas numa estrutura de dados associada ao próprio arquivo. Esta estrutura é chamada nó-i, é a forma como o Linux implementa compartilhamentdo arquivo.

5.4 - Estrutura do Sistema de arquivos do LINUX Release 1.2

5.4.1 - Apresentação

Este trabalho é baseado na versão 1.2 da Estrutura do Sistema de arquivos do LINUX (**LINUX File System Structure**) **FSSTND,** que por sua vez é baseado em um documento de consenso da comunidade Linux (que poderá ser encontrado na internet -**www.linux .org**) o layout do sistema de arquivos foi inicialmente desenvolvido dentro da lista de e-mail FSSTND do **LINUX-ACTIVISTS**.
O coordenador do FSSTND é Daniel Quinlan **<Daniel.Quinlan@linux.org>.** Uma parte considerável deste trabalho foi tirado da **FAQ** (Lista de perguntas mais frequentes) mantida por Ian McCoghrie (**FSSTND-FAQ**). Este documento está

disponível via ftp anonymous em tsx-11.mit.edu no diretório /pub/linux/docs/linux-standards/fsstnd/ FSSTND-FAQ

Nosso trabalho enfocará a estrutura do sistema de arquivos para LINUX típico, incluindo a localização de arquivos e diretórios, e o conteúdo de alguns arquivos de sistema.

5.4.2 - Características Sistema de Arquivos

O sistema de arquivos Linux esta caracterizado por:

- Uma estrutura hierárquica.
- Um tratamento consistente da informação dos arquivos.
- Proteção dos arquivos.

O sistema de arquivos Linux segue o mesmo princípio básico que a maioria dos sistemas de arquivos UNIX seguem. Apesar que o sistema de arquivo não concordar em 100% com cada aspecto possível de alguma implementação particular do sistema UNIX. De qualquer forma, muitos dos aspectos da implementação do sistema de arquivos estão baseados em idéias encontradas em sistemas similar ao UNIX system V, outros fatores também foram levado em conta tais como :

- Práticas comuns na comunidade LINUX.
- A implementação de outras estruturas de sistemas de arquivos.
- Definição das categorização ortogonal de arquivos: Compatível vs. não compátivel. e variável vs. estáticos.

A informação compatível é aquela que pode ser compartida entre várias máquinas diferentes. A não compatível é aquela que deve ser localizada em uma máquina particular. Por exemplo, os diretórios local dos usuários são compatível, porém os arquivos de bloqueio do dispositivo (lock file) não são compatíveis.

A informação estática inclui arquivos, biblilotecas, documentação e tudo aquilo que não precisa da intervenção do administrador do sistema. A informação variável é tudo aquilo que se troca com a intervenção do administrador.

O entendimento destes princípios básicos ajudará a guiar a estrutura de qualquer sistema de arquivos bem planejado.

A distinção entre informação compatível e não compatível é necessária por várias razões:

- Em um ambiente de rede, existe uma boa quantidade de informação que se pode compartilhar entre diferentes máquinas para o aproveitamento de espaço e facilitar a tarefa da administração.
- Em um ambiente de rede, certos arquivos contém informação especifica de uma só máquina, portanto, estes sistemas de arquivos não podem ser compartilhados (antes de tomar medidas especiais).
- As implementações de fato do sistema de arquivos não nos permitem que a hierárquia /usr fosse montada somente para leitura, porque possuía arquivos e diretórios que necessitavam ser escritos muito freqüentemente. Este é um fator que deve ser atacado quando algumas parte de /usr são compartilhadas na rede.

A distinção "compatível" pode ser usada para suportar, por exemplo:

- uma partição /usr (o componente de /usr) montada (somente para leitura) através da rede (usando NFS).

- uma partição /usr (o componente de /usr) montada somente para leitura de um cd-rom, pode ser considerado como um sistema de arquivos somente para leitura, compartilhado com outros sistemas LINUX utilizando o sistema de e-mail como uma rede.

A distinção "estática" contra "variável" afeta o sistema de arquivos de duas maneiras principais:

- Arquivo da /(raiz) contém ambos tipos de informação, variável e estática necessita permitir leitura e escrita.
- Arquivo do /usr tradicional contém ambos tipos de informação variável e estática e os arquivos poderíamos desejar montar-los somente para leitura, é necessário proporcionar um método para fazer que /usr funcione somente para leitura. Isto se faz com a criação de uma hierárquia /var que funciona leitura e escrita (ou é parte de uma partição leitura-escrita tal como /), que diminui muito a funcionalidade tradicional da partição /usr.

O diretório raiz / . Esta sessão descreve a estrutura do diretório raiz. O conteúdo do sistema de arquivos raiz será adequado para carregar, dar boot, restaurar, recuperar o sistema:

Para carregar o sistema, deve estar presente o suficiente como para montar /usr e outras parte não essenciais do sistema de arquivos.

Esta inclui ferrramentas, informação do configuração e do boot de inicialização (boot loader) e algum outra informação essenciais ao inicilizar.

Para habilitar a recuperação e /ou a reparação do sistema, estará presente no sistema de arquivos raiz aquelas ferrramentas que o administrador experimentado necessitaria para diagnosticar e reconstruir um sistema danificado .

Para restaurar um sistema, estaram presente no sistema de arquivos raiz aquelas ferrramentas necessárias para restaurar o sistema em floppy , fitas, etc.

A principal preocupação e que procuraremos colocar muito pouca coisas do sistema de arquivos raiz, é a meta de manter / tão pequeno quanto possível. Por várias razões é desejável mante-lo pequeno.

É freqüentemente montado em mídia muitas pequenas. Por exemplo muitos usuários de LINUX instalam e recuperam sistemas montando como o disco copiado de disquete 1.44Mb.

O sistema de arquivos / tem muitos arquivos de configuração especificos de um sistema. Possíveis exemplos são de um kern, que é específicos do sistema, um hostname diferente, etc. Isto significa que o sistema de arquivos / não é sempre compatível entre sistemas em rede. Mantendo-o pequeno no sistemas de rede, se minimiza o espaço perdido no servidor utilizados pelos arquivos não compatível. Também permite estações de trabalho com winchesters locais menores.

Os erros do disco, que corrompe as informação no sistema de arquivos / são um problema maior que os erros em qualquer outra partição. Um sistema de arquivos / pequeno é menos propenso a perder arquivos como resultado de uma falha do sistema.

É recomendável colocar principalmente os arquivos em /etc/mtab. De qualquer forma, não se necessita que o sistema de arquivos / esta totalmente armazenado localmente. A partição / não tem porque estar armazenada localmente pode ser especificada pelo sistema por exemplo, poderiam estar montada de um servidor NFS.

O software não deverá criar ou pedir arquivos de subdiretórios especiais. A estrutura do sistema de arquivos LINUX proporciona suficiente flexibilidade para qualquer pacote.

5.4.3 - Composição dos Diretórios

/ --- O Diretório Raiz

A composição do diretorio raiz de um sistema Linux típico pode ser representado pela Tabela 01.

Tabela 01	
bin	arquivos executáveis(binários) de comandos essenciais pertencentes ao sistema e que são usados com freqüencia.
boot	arquivos estáticos de boot de inicialização(boot-loader)
dev	arquivos de dispositivos de entrada/saída
etc	Configuração do sistema da máquina local com arquivos diversos para a administração de sistema.
home	diretórios local(home) dos usuários.
lib	arquivos da biblilotecas compartilhadas usados com freqüencia
mnt	Ponto de montagem de partição temporários
root	Diretório local do superusuário (root)
sbin	Arquvios de sistema essenciais
tmp	arquivos temporários gerados por alguns utilitários
usr	todos os arquivos de usuários devem estar aqui (segunda maior hierárquia)
var	Informação variável

Cada diretório listado será discutido em detalhe em uma subsessão separada mas adiante. /usr e /var, cada un tem seu própria sessão de documentos.

O kern do LINUX estaria localizado na raiz / ou no /boot. Se estiver localizado em / é recomendado usar O nome VMLINUX o VMLINUZ, nome que deverá ser usados em programas fonte do kern do LINUX recentemente. Mais informação da localização do kern pode-se encontar na sessão sobre a raiz / neste trabalho.

5.4.3.1 - Subdiretório /bin

- Composição : Arquivos Binários de comandos essenciais de usuários (disponíveis para todos os usuários).

Contém os comandos que podem ser utilizados pelos usuários e pelo administrador do sistema, porém que são requeridos no modo mono-usuário (single-user mode) pode também conter comandos que são utilizados indiretamente por alguns scripts.

Todos os arquivos utilizados somente pelo root, tal como **daemons, init, gette, update**, etc. Estariam localizados em /sbin ou /usr/sbin dependendo se são ou não essenciais. Não abra subdiretórios dentro /bin.

Os arquivos dos comandos que não são suficientemente essenciais para estar em /bin estaram localizados em /usr/bin, os elementos que são utilizados pelos usuários isoladamente (porém não pelo root) (**mail, chsh, etc**.) não são suficientemente essenciais para estar dentro da partição /.

5.4.3.1.1 - Arquivos e/ou comandos disponíveis em bin

Os arquivos que devem estar em /bin são : comandos gerais e comandos de rede.

Comandos gerais:

Os seguintes comandos deverão sido incluídos porque são essenciais.
Alguns estaram presente e tradicionalmente deverão estado em /bin.

- {**arch, cat, chgrp, chmod, chown, cp, date, dd, df, dmeg, echo, ed, false, kill, in, login, mkdir, mknod, more, mount, mv, ps, pwd, rm, rmdir, sed, setserial, sh, sfte , seu, sinc, true, umount, uname**}.

Se /bin/sh é bash, então /bin/sh seria em links simbólico a /bin/bash dado que bash se comporta diferente quando é carregado como sh ou bash. A pdksh que pode ser a /bin/sh nos discos de instalação e seria igualmente carregada a que /bin/sh faz um links simbólico a /bin/ksh. Outros links simbólicos de sistemas utilizando outros programas, então a partição / conterá os componente mínimos necessários.

Por exemplos, muitos sistemas incluiria cpio como de segunda utilidade mais é usado para reparos depois do tar. Porém jamais se espera restaurar o sistema da partição /, então estes arquivos podem ser omitidos (montar / em chip ROM, montar /usr no NFS). Se a restauração do sistema se planeja através da rede, Então FTP o TFTP (junto com todo o necessário para obter uma conexão FTP) estariam disponíveis na partição /.

Os comandos de restauração podem aparecer em /bin ou /usr/bin em sistemas LINUX diferentes.

Comandos de redes.

Estes são unicamente os arquivos de rede que os usuários e o root queiram ou necessitem executar que não estejam no /usr/bin ou /usr/local/bin {**domain name**, **hostname**, **netstat**, **ping**}.

5.4.3.2 - Subdiretório /boot:

- Composição : arquivos estáticos do boot de inicialização (boot loader).

Este diretório contém tudo que é necessário para carregar o sistema, exceto os arquivos de configuração e o gerenciador de boot. O /boot é utilizado para qualquer coisa que se utiliza antes do kernel execute /sbin/init. Este inclui setores master de inicialização (master boot sectors) guardados, arquivos de mapa de setor e qualquer outra coisa que não é editada manualmente. Os programas necessários para consertar o boot de inicialização e capaz de carregar um arquivo (tal como o gerenciador de boot [lilo]) estaram localizados em /sbin. Os arquivos de configuração para carregar de inicialização poderiam estar localizados em /etc.

Como o exposto acima, o kern do LINUX pode estar localizado em / ou /boot, se estiver em /boot, é recomendado usar um nome mais descritivo.

5.4.3.3 - Subdiretório /dev

- Composição : arquivos de dispositivos de entrada/saída.

Este é diretório dos dispositivos. Contém um arquivo para cada dispositivo que o kern do LINUX pode suportar.

/dev também contém. um script carregado **MAKEDEV** o qual pode criar dispositivos quando necessitar. Pode conter um **MAKEDEV** local para dispositivos locais.

MAKEDEV deve fazer previsão para criar qualquer arquivo de dispositivo especial listado na lista de dispositivos suportados pelo Linux.

Os links simbólicos não devem ser distribuidos no sistemas LINUX, somente como preve na lista de dispositivos de LINUX. Isto é porque as instalações locais seguro diferem daquelas da máquina do administrador. Além disso, um script de instalação configura links simbólicos na instalação, estes ligação seguramente não se atualizaram se houverem trocas locais no hardware. Quando utilizados responsavelmente são de bom uso.

A lista de dispositivos compatível com o LINUX, é mantida por Peter Anvin (**peter.anvin@linux.org**). Estes arquivos especial de dispositivo estão disponível no endereço eletrônico da internet **ftp.eggdrassiml.com**, no diretório **/pub/device-list**.

5.4.3.4 - Subdiretório /etc

- Composição : Configuração do sistema da máquina local com arquivos diversos para a administração de sistema.

Contém arquivos e diretórios que são locais ao sistema atual.

Nenhum arquivo deve ir diretamente dentro /etc. Os arquivos que nas primeiras versões encontravam-se em /etc, irá em /sbin ou /usr/sbin. Isto inclui arquivos tal como init, gette e update, arquivos como hostname que são utilizados por usuários comuns e pelo root não iriam em /sbin seriam em /bin.

Subdiretórios de /etc

Tipicamente /etc possui dois sudiretórios :

- X11 arquivos de configuração para Ox11
- sk Esqueletos da configuração usuários

O /etc/sk é a localidade para os chamados arquivos esqueletos de usuários, que são os dados por default quando um novo usuário recebe uma conta, esta diretório pode conter subdiretórios para diferente grupos de usuários (/etc/skell/apoio, /etc/skell/usuários).

O /etc/X11 é o lugar recomendado para todos os arquivos de configuração X11 locais da máquina. Este diretórios é necessário para permitir o controle local se /usr for colocado somente para leitura. Os arquivos que devem ir neste diretório incluem **Xconfig** (e /o XF86Config) e **Xmodmap**. O /etc/X11/xdm contém os arquivos de configuração xdm. Estes são a maioria dos arquivos normalmente gravados em /usr/lib/X11/xdm; veja /var/lib/xdm, para maior informação.

5.4.3.4.1 - Arquivos e/ou comandos disponíveis em /etc

Este descrição do conteúdo é generica, portanto não está totalmente completa, pode haver algumas variações dependendo do distribuidor do Linux ou do administrador de sistema. Os arquivos /etc são composto de :

- Arquivos gerais;
- Arquivos de rede.

Os arquivos gerais são necessários na maioria dos sistemas LINUX, tais como :

{adjtime, csh.login, disktab, fdprm, fstab, gettedefs, group, inittab, issue, ld.so.conf, lilo.conf, magic, motd, mtab, mtools, passwd, profile, psdaTabelase, securette , shells, se sog.conf, tercamp, tte te pe}

Os arquivos de Rede mais utilizados na maioria dos sistemas LINUX são :

{exports, ftpusers, gateway, hosts, host.conf, host.equiv, host.lpd,

inetd.conf, networks, printcap, protocols, reolv.conf.rpc, service}

Existe dois modo para a instalação dos scripts de comandos "rc" os quais são chamados por init no momento de carregar, ou o modo /etc/rc.d/* etc do System V. Qualquer que seja o escolhido pode ser utilizado uma mescla dos dois.

Os sistemas com a senha de passwords sombreadas (shadow password) terão arquivos de configuração adicionais, em /etc (/etc/shadow e outros) e /usr/bin (useradd, usermod, e outros).

5.4.3.5 - Subdiretório /home:

Composição : diretórios locais dos usuários (opcional)

O sudiretório /home é claramente um sistema de arquivos específico do diretório local. A regra de criá-lo difere de máquina para máquina. Descreve uma localização sugerida para os diretórios local dos usuários, assim, recomendamos que todas as distribuições LINUX usem esta lugar como localização default dos diretórios locais.

Em sistemas pequenos, cada diretório de usuário é um dos subdiretórios debaixo /home, /home/dirson, /home/raulison, /home/weslei, etc.

Em sistemas grande (especialmente quando os diretórios /home são compartilhados entre várias máquinas usando NFS) é útil subdividir os diretórios local . A subdivisão pode ser implementada utilizando subdiretórios tal como /home/apoio, /home/docs, /home/cartas, etc. Muitas pessoas preferem colocar as contas dos usuários numa variedade de lugares, portanto, ninguém deverá confiar nesta localização. Se você deseja encontar o diretório local de qualquer usuário, deveria usar a função de biblilioteca **getpwent** em vez de contar com /etc/passwd, porque a informação pode estar armazenada remotamente usando sistemas como NIS.

5.4.3.6 - Sudiretório /lib:

- Composição : Biblilotecas compartilhadas e módulos do kern essenciais.

O diretório /lib contém aquelas biblilotecas compartilhadas que são necessária para carregar o sistema e executar os comandos do sistema de arquivos raiz.

Módulos -> Módulos de kern carregavéis.

Estes incluem em /lib/libc.so.*, /lib/libm.so.*, O linkador dinâmico compartilhado /lib/ld.so.*, e outras biblilotecas compartilhadas requeridas por arquivos em /bin e /sbin.

As biblilotecas que são necessárias somente pelos arquivos /usr (como qualquer arquivo X Windows) não pertencem a /lib. Só as biblilotecas compartilhadas requeridas para executar os arquivos dentro de /bin e /sbin devem estar ali. A biblilioteca libm.so.* poderia estar localizada em /usr/lib se não é utilizada de nenhuma forma em /bin ou /sbin.

Por razão de compatibilidade, /lib/cpp necessita existir como uma referência ao pre-processador C instalado no sistema. A localização usual do arquivo é /usr/lib/gcc-lib/ <target>/<versão>/cpp. Pode existir links /lib/cpp apontando para estes arquivo ou a qualquer outra referência a ele que exista no sistema de arquivos. (Por exemplo, /usr/bin/cpp é utilizado frequentemente). A especificação para /lib/module ainda não foi definida, pois ainda não há um concenso na comunidade

Linux.

5.4.3.7 - Subdiretório /mnt

- Composição : Utilizados para armazenamento de arquivos montados temporariamente.

Este diretórios foi previsto para o administador poder montar temporariamente sistemas de arquivos quando necessitar. O conteúdo deste diretório é um assunto local e não deve afetar a maneira que executamos nenhum programa. É recomendamos a não utilização deste diretório para programas de instalação, e sugerimos utilizar um diretório temporário adequado que não esta em uso pelo sistema.

5.4.3.8 - Subdiretório /proc

- Composição : Sistema de arquivos virtual de informação de processos do kernel.

O sistema de arquivos proc é utilizado para manipular informação de processos e de sistema em vez de /dev/kmem e outros métodos similares. É recomendado a utilização deste para o armazenamento e obtenção de informação de processos, assim como outras informação do kern ou da memória.

5.4.3.9 - Subdiretório /root (opcional)

Composição : Diretório local do superusuário (root)

O diretório / é tradicionalmente o diretório local do usuário root nos sistemas UNIX. /root utiliza-se em muitos sistemas LINUX e em alguns sistemas UNIX. O diretório local da conta do usuário root pode ser determinada por preferências locais. As possibilidades óbvias inclue em /, /root, e /home/root. Se o diretório local do root não esta armazenado na partição raiz, será necessário assegurar-se que tome / por default caso não seja localizado.

Não é recomendado o uso da conta root para coisas corriqueiras tal como ler o e-mail e ver as notícias (mail & news), recomendá-se que seja usada somente para a administração do sistema. Por esta razão recomendamos que não apareçam subdiretórios como Mail e News no diretório local da conta do usuário root. É recomendado que o mail para root seja redirecionados a um usuário mais adequado.

5.4.3.10 - Subdiretório /sbin:

- Composição : arquivos de Sistema (algumas vezes mantidos em /etc)

Os utilitários usados pela administração do sistema (e outros comandos que somente o root utiliza) estão armazenados em /sbin, /usr/sbin, e /usr/local/sbin. /sbin, tipicamente contém arquivos essenciais para dar boot ao sistema, além dos arquivos em /bin. Qualquer coisa que se executar depois sabendo que /usr foi montado (quando não há problemas) deveria estar em /usr/sbin. Os arquivos da administração

do sistema root local devem estar em /usr/local/sbin.

Decidir que arquivos vão no diretório de /sbin é difícil. Se o usuário necessitar executá-lo, deve de ir para outro diretório. Se somente o administrador do sistema ou o root necessitem executar, tais como scripts da administração, então deve ir em /sbin (não /usr/sbin ou /usr/local/sbin, se o arquivo não é vital para a operação do sistema).

Arquivos como **chfn** que os usuários usam ocasionalmente devem estar em /usr/bin. **ping** que é absolutamente necessário para o root é também frequentemente usado pelos usuários, deverão estar em /bin. Os usuários comuns não terão que por os diretórios sbin em seu caminho (path).

É recomendado que os usuários tenham permissão de leitura e execução em tudo que se encontra em /sbin exceto talvez certos programas; **setuid** e **setgid**. A divisão entre /sbin e /bin não foi criada por motivos de segurança para evitar que os usuários violem o sistema operacional, foi cirada para promover uma boa partição entre arquivos que todos usam e os que são utilizados principalmente para as tarefas administrativas. Não há utilidade inerente na segurança em fazer que /sbin esteja fora do alcance dos usuários.

5.4.3.10.1 - Arquivos e/ou comandos armazenados em /sbin

Arquivos armazenados são dos seguintes tipos :

- comandos gerais;
- comandos de saída;
- comandos de manipular sistema de arquivos;
- gerenciador de boot de inicialização e
- comandos de rede.

Os comandos gerais são **clock, gette, init, update, mkswap, swapon, swapoff, telinit.**

Os comandos de saída são **fastboot, fasthalt, halt, reboot, shutdown.**

Os comandos de manipular sistemas de arquivos são **fdisk, fsck, fsck.*, mkfs, mkfs.*,** onde * = é um dos seguinte. **ext, ext2 minix, msdos, xia,** e talvez outros.

Os comandos do sistema de arquivos ext2 (opcional) são **badbocks, dumpe2fs, e2fsck, mke2fs, mkost+found, tune2fs.** O Gerenciador do boot de inicialização (lilo) e os
comandos de Rede : **arp, ifconfig, route.**

5.4.3.10.2 - Arquivos opcionais em /sbin:

Os arquivos estáticos (compilados estaticamente) estão o **sln** e **nc** estático, **nc** são utilitários quando ocorrem erros. O principal uso do **sln** (reparar links simbólicos incorretos em /lib depois de uma atualização mal sucedidas) não é a preocupação maior, já que existe o programa **ldconfig** (usualmente localizado em /usr/sbin) e pode atuar como um assistente guiando para atualizar as biblilotecas dinâmicas. O **nc** estático é útil em algumas ocasiões de emergência. Note que estes não necessitam ser obricatoriamente versões compiladas estaticamente dos **ln** e **nc**, porém podem ser compilados estáticamente.

O arquivo **ldconfig** é opcional em /sbin, dado que um usuário pode escolher executar **ldconfig** ao dar boot, em vez de só quando atualizam as biblilotecas compartilhadas . (Não está claro se é ou não vantajoso executar **ldconfig** em cada inicialização). Assim, alguns que gostam de ter **ldconfig** a mão na situação que se tem um **sln**, pois não se sabe como nomear os links**.**

5.4.3.11- Subdiretório /tmp

Composição : arquivos temporários gerados por alguns arquivos utilitários.

O /tmp é utilizado para arquivos temporários, preferencialmente em dispositivo rápido (um sistema de arquivos basado em memória por exemplo). A "permanência" da informação que é armazenada em /tmp é diferente de aquela que é armazenada em /var/tmp. /tmp pode ser limpo em cada inicialização ou a intervalos relativamente frequentemente. Portanto, não se deve operar a informação armazenada em /tmp permanecendo por algum período determinado de tempo.

Os programas devem utilizar /tmp ou /var/tmp (que era originalmente /usr/tmp) de acordo os requisitos esperados da informação, pois não devem colocar nenhum arquivo particular em qualquer diretório de armazenamento temporário.

Os administradores de sistemas podem querer ajuntar /tmp a algum outro diretório, tal como /var/tmp. Isto é útil, por exemplo, para conservar espaço na partição raiz. Se está é executada, então a permanência de arquivos em /var/tmp deve ser mesmo tão grande como a de /tmp. O subdiretório /tmp pode estar na memória RAM, /var/tmp nunca poderá se localizar-se em algum dispositivo RAM.

5.4.3.12 - A hierárquia /usr.

O subdiretório /usr é a segunda maior seção do sistema de arquivos. /usr é informação compartilhada, somente de leitura, isto significa que /usr, deve ser compartilhada entre várias máquinas que utilizam o LINUX e não deve exibir qualquer informação local de uma máquina ou que varia com o tempo, deve ser armazenada em outro lugar.

Nenhum pacote grande (como **TeX** o **GNUEmacs**) deve utilizar o subdiretório direto abaixo de /usr, em vez disso, deve haver um subdiretório dentro /usr/lib (o /usr/local/lib caso tenha sido instalado localmente), a propósito, como o sistema **X Windows** faz-se uma exceção permitindo um considerável precedente e a prática amplamente aceita. Exemplo de um subdiretório /usr típico :

```
carvalho:/usr$ ls
X11@          etc/          lib/          spool@
```

```
X11R6/          games/              local/         src/
X386@           i486-linux/         man/           tclX/
adm@            i486-linuxaout/     openwin/       tkX/
bin/            i486-sysv4/         preserve@      tmp@
dict/           include/            sbin/
doc/            info/               share/
carvalho:/usr$
```

5.4.3.12.1 - Subdiretórios /usr (permanente)

Em um sistema típico teremos mais ou menos os seguintes diretório :

X11R6	Sistema X Windows Versão 11 release 6
X386	Sistema X Windowss Versão 11 release 5 em plataformas X 386
bin	A maioria dos comandos de usuário
dict	Listas de palavras
doc	Documentação miscelânia
etc	Configuração do Sistema
games	Jogos e arquivos educacionais
include	arquivos header(cabeçalhos) incluidos por programas C
info	Diretório primário, o sistema GNU Info
lib	Biblilotecas
local	Hierárquia local
man	Manual on line
sbir	Arquivos de Administração do Sistema não vitais
share	Informação independente da arquitetura
src	Código fonte

Os seguinte links simbólicos a diretórios podem estar presentes. Esta possibilidade baseia-se na necessidade de preservar a compatibilidade com sistemas anteriores haja visto que em todas as implementações pode asumir o uso da hierarquia /var, poderão existir os seguintes links :

```
/usr/adm             ------------------>     /var/adm
/usr/préerve         ------------------>     /var/préerve
/usr/spool           ------------------>     /var/spool
/usr/tmp             ------------------>     /var/tmp
/var/spool/locks     ------------------>     /var/lock
```

Uma vez que o sistema não precise mais de alguns dos liks anteriores, deve existir um links deste /usr/X11 apontando para a hierarquia do sistema X Windows atual.

5.4.3.12.2 - Subdiretório /usr/X386

É composta do sistema X Windows, versão 11 release 5 em plataformas X 86, esta hierarquia é geralmente idêntica a /usr/X11R6, exceto que os links simbólicos de /usr devem estar ausente se estiver instalado /usr/X11R6.

5.4.3.12.3 - Subdiretório /usr/bin

É composta da maioria dos comandos do usuário, este é o diretório principal de comandos executáveis no sistema possui o **mh** (comandos para o sistema de manipular e-mail MH) e o **X11** (link simbólico até /usr/X11R6/bin).

Os interpretadores de scripts dos shell (invocados com **#! <rota>** na primeira linha do script de shell) não podem depender de uma rota, é vantajoso o padronizar a localização dos elos. A shell Bourne e C é tão fixos em **/bin**, pois **Perl, Pethon, Tlc** se encontram em muitos lugares diferentes **/usr/bin/perl, /usr/bin/pethon e /usr/bin/tcl** devem referênciar a os interpretador de shell perl, pethon e tcl respectivamente. Estes podem ser links simbólicos a localização física dos interpretador da shell.

5.4.3.12.4 - Subdiretório /usr/dict - Listas de palavras

Arquivos recomendados em /usr/dict (words), tradicionalmente esta diretório contém somente arquivos words de palavras inglesas, o qual é utilizado por "look" para vários programas de ortografia, words pode utilizar ortografia americana ou britânica. Os usuários que precisam ambos, podem concatenar words a /usr/dict/american-english ou /usr/dict/ british-english.

As listas de palavras para outros linguagem pode usar o nome em inglês para a linguagem, por exemplo, /usr/dict/french, /usr/dict/danish, etc. Estes devem, se possível, utilizar o jogos de caracteres ISO 8859 que faz aprópriado para linguagem em questão, e se possível, o jogo de caracteres ISO 8859-1 (Atin1) deve ser utilizado (muito raramente é possível fazê-lo)

Qualquer outra lista de palavras, tal como o diretório **web2**, deve ser incluido aqui,

É razoável ter aqui só as listas de palavras, e que elas são os únicos arquivos comum a todos os verificadores de ortografia.

5.4.3.12.5 - Subdiretório /usr/etc

Contém a configuração do sistema, porém armazenar a configuração /usr/etc do software que se encontra em /usr/bin e /usr/sbin é um problema. Montar /usr somente para leitura de um CD-ROM ou através de NFS é difícil no melhor dos casos.

Uma possível solução que considerada foi eliminar completamente /usr/etc e especificar que todas as configurações se armazenem em /etc. Acontece que existe a possibilidade de que muitos site podem querer ter alguns arquivos de configuração que não estejam na sua máquina local.

Eventualmente, decide-se que /etc deverá ser o único diretório que seja referenciado pelos programas (Isto é, todos devem buscar configurações em /etc e não /usr/etc). Qualquer arquivo de configuração que necessissário para todo o sistema e que não era necessário antes de montar /usr (em uma situação de emergência deve estar localizado em /usr/etc). Arquivos especificos em /etc, em máquinas específicas podem ter ou não um link simbólicos aos arquivos de configuração localizados em /usr/etc. Isto também significa que /usr/etc é tecnicamente um diretório opcional no sentido restrito, mesmo assim recomendamos que todos os sistemas LINUX o incorporem.

Não é recomendado que /usr/etc contenha links simbólicos que apontem para arquivos /etc. Isto é desnecessário e interferem no controle local das máquinas que

compartem o diretório/usr.

5.4.3.12.6 - Subdiretório /usr/include

Neste diretório é onde todos os arquivos include de uso geral do sistema para programação em linguagem C e C++ podem ser localizados.

Descrição dos principais sudiretórios de /usr/include :
/usr/include   arquivos include
X11   Link simbólico até /usr/X11R6/include/X11
arpa   Definição do protocolo definido por ARPNET.
asm   Link simbólico até /usr/scr/linux/include/asm-<arch>.
bsd   arquivos include de compatibilidade com BSD.
g++   arquivos include de GNU C++.
gnu   arquivos include GNU.
linux   Link simbólico a /usr/src/linux/include/linux.
net   Definição genéricas relacionadas com rede.
netax25   Definição específicas a +AX25 (ARRL AX25).
Netinet Definição específicas a TCP/IP.
netipx Definição específicas a +IPX (NovOIPX/SPX).
protocols   Definição de protocolos(basadas em INET)
readline   A biblilioteca readline GNU.
rpc   Definição RPC de Sun Microsystem.
Rpcsvc Definição de serviços RPC de Sun Microsystem.
sys   arquivos include de geração de sistemas

O subdiretório **arpa** contém definições de header de protocolos para os protocolos **ARPANET, TCP/IP**, definições para ftp, prototipos telnet e material similar.
O subdiretório **net** contém definições genéricas relacionadas com a rede, define a interface sistema vs. kern, detalhes da família de protocolo, etc.
O subdiretório **netinet** contém definições específicas de INET (DARPA Internet, que também é contida no TCP/IP)
ARRL AX.25 é melhor conhecido como pacote de transmissão via radio (packet radio). Os protocolos novell **IPX/SPX** são parte dos serviços de arquivos Novell Netware.

5.4.3.12.7 - Subdiretório /usr/lib

Inclui as biblilotecas para programas e pacotes, inclue as biblilotecas objeto, arquivos de programa compilador, informação estática de várias casos, ambos, códigos executável (por exemplo os arquivos internos de **gcc** estão localizados abaixo **/usr/lib/gcc-lib**) e outros tipos de informação.

/usr/lib/ - biblilotecas para programação e pacotes:

X11   Link simbólico para /usr/X11R6/lib/X11
emacs arquivos de suporte estáticos para o editor GNUEmacs.
games arquivos de dados estáticos para /usr/games.
groff   Biblilotecas / diretórios para GNU groff
gcc-lib arquivos/diretórios especificos do sistema para gcc.

kbd Tabelas de tradução de teclado e informação relacionada.
Mh Biblilotecas para o sistema de manipular e-mail MH:
news Cnews/INN.
smail Smail.
terminfo diretórios para a base de dados terminfo.
texmf TeX/MF (e ATeX) biblilotecas de informação.
uucp Comandos de UUCP.
zoneinfo Configuração e informação da zona horaria.

Historicamente, /usr/lib é incluido além disso alguns comandos executáveis tais como **sendmail** e **makewhatis**.

Dado que **makewhatis** não é referênciado por outros programas, não há problemas ao mover para um diretório de arquivos executáveis. Arquivos que os usuários precisam para usar **makewhatis**, /usr/lib de onde pertencem. O arquivo **catman** que repassa ao script **makewhatis** em muitos sistemas LINUX, deve também estar em usr/bin.

O arquivo **sendmail** é referênciado por muitos programas com seu nome histórico /usr/lib/sendmail. Este deve ser um links simbólico, a localização layout para os agente de transferência de e-mail com uma interface de linha de comando compatível com o sendmail, /usr/bin/sendmail.

Em sistemas que utilizam smail devem localizar smail em /usr/sbin/smail e /usr/bin/sendmail deve ser um links simbólico a smail.

Este regra vai de encontro também com a nova ocorrência no layout sendmail definida em Sendmail 8.6.x e BSD 4.4.

Note que esta localização requer que /usr/sbin e /usr/sbin/sendmail devem ser executáveis para usuários normais.

Qualquer pacote de programa que contenha e precisa informação que não necessite ser modificada deve armazenar tal informação em /usr/lib (o /usr/local/lib, esta instalado localmente). Recomenda-se a utilização de um subdiretório em /usr/lib para este propósito.

A informação de jogos armazenada em /usr/lib/games deve ser apenas informação estática. Qualquer arquivo modificável tal como arquivos demarcado, registros de jogos e similar, devem de ser localizados em var/lib. É necessário para compatibilidade de jogos, pode-se usar um links simbólico desde /usr/games/lib até /usr/lib/games.

Nota: nenhuma informação especifica de host para o sistema X Windows deve armazenar-se em /usr/lib/X11 (que é realmente /usr/X11R6/lib/X11). Os arquivos de configuração especificos do host tal como **Xconfig** o **XF86Config** devem ser armazenados em /etc/X11. Este deve incluir informação de configuração como o stem.twmrc, se for somente um links simbólico, um arquivo de configuração mais global (talvez em /usr/etc/X11 ou /usr/X11R6/lib/X11).

5.4.3.12.8 - Subdiretório /usr/local

A hierarquia /usr/local está para ser utilizada pelo administrador de sistemas quando instala o Linux localmente. Necesssita ficar a salvo de ser sobrescrito quando o software do sistema se atualiza. Pode ser usado por programas e por informação que são compatível entre um grupo máquinas, pois não se encontram em /usr.

/usr/local Diretórios da Hierárquia local
bin arquivos
doc Documentação local

etc   arquivos de configuração utilizados somente no local
games Jogos instalados localmente
lib   Biblilotecas para /usr/local
info   Páginas de informação local
man   Hierárquias de páginas de manual para /usr/local
sbin   Administração do sistema
scr   Código fonte local.

Este diretório deve estar vazio ao terminar de instalar LINUX pela primeira vez. Não deve haver exceções regra , exceto quiça os subdiretórios vazios listados.

O software instalado localmente deve estar localizado dentro de /usr/local, em vez de /usr a menos que esteja sendo instalado para reemplantar ou atualizar software em /usr.

Note que o software localizado em / ou em /usr pode ser sobrescrito por atualizações do sistema (assim mesmo, é recomendado que as distribuições não sobrescrevam informações /etc fora destas circunstâncias). Por esta razão, o software local não deve se colocado fora de /usr/local sem uma boa causa.

5.4.3.12.9 - Subdiretório /usr/man

Inclui as paginas do manual, detalha a organização das páginas do manual através do sistema, devem estar dentro de /usr/man.

As páginas do manual estão armazenadas <mandir>/<locais>/man [1-9]. Faremos uma pequena listagem de <mandir> e <locais> :

<mandir>/<locais>   uma hierarquia de páginas de manual.
man1   Programas para usuários.
man2   Chamadas do sistema.
man3   Subrotinas e funções de biblioteca.
man4   Dispositivos.
man5   Formatos arquivos.
man6   Jogos.
man7   Misceláneas.
man8   Administração do Sistema.
man9   Funções e variáveis internas do kernel.

O <mandir> primário do sistema é /usr/man contém informação do manual para comandos e informação abaixo dos sistemas de arquivos / e /usr. Obviamente não há páginas de manual em / porque não se necessitam para carregar nas emergências.

Deve-se fazer reserva na estrutura de /usr/man para o suporte de páginas do manual que estão escritas em diferentes idiomas (multiplos idiomas). Estas providências devem levar em conta o armazenamento e referência destas páginas do manual. Os fatores relevantes incluir no idioma (inclue diferenças basedas na geografia) e código do conjunto caracteres.

Esta nomenclatura dos subdiretórios de idiomas de /usr/man esta basada no apêndice e do padrão POSIX 1003.1 que descreve a cadeia de identificação locais.

O método mas aceito para descrever um ambiente cultural. A cadeia <locais> é:

- **<idioma>[<_territorio>][.<conjunto_de_caracteres>][,<versão>]**

O campo <idioma> vem do ISO639 (um código para a representação dos nomes dos idiomas). Seja os caracteres especificado no padrão ISO, com minúsculas somente.

O campo <_territorio> será o código das letras de ISO3116 (uma especificação da representação dos nomes dos países, se possível (muita gente está familiarizada com o código 2 letras espelhado no código país como e-mail).

O campo <conjunto_de_caracteres> deve representar o layout que descreve o código caracteres. Se o campo <conjunto_de_caracteres> é só uma especificação numérica, o número representa o número do layout internacional que descreve o conjunto caracteres. Recomenda-se que utilizar uma representação numérica, sempre que for possível (especialmente o padrão ISO), que não inclua símbolos de pontuação e que todas as letras sejam minúsculas.

Um parâmetro que especifique <versão> do perfil pode ser colocada depois do campo <conjunto_de_caracteres >. Esta pode utilizar-se para diferenciar as necessidade culturais.

Em sistemas que usem só um idioma e um código do conjunto de caracteres para todas as páginas do manual, pode-se omitir a subcadeia <locais> e armazenar todas as páginas do manual em <mandir>. Por exemplo no sistemas que só tem páginas do manual em inglês codificados na ASCII, podem armazenar as páginas do manual (Os diretórios man[1-9]) diretamente em /usr/man.

Em países nos quais existe um código do conjunto caracteres no layout, pode omitir o campo <conjunto_de_caracteres>, porém é bastante recomendado que a inclua, especialmente para países com vários layouts.

Exemplos de vários manuais encontrados :

Idioma	Países	Conjunto caracteres	Diretório
Inglês	--------	ASCII	/usr/man/em
Inglês	Reino Unido	ASCII	/usr/man/em_GB
Inglês	Estados Unidos	ASCII	/usr/man/em_US
Francês	Canadá	ISO8859-1	/usr/man/fr_CA
Francês	França	ISO8859-1	/usr/man/fr_FR
Alemão	Alemanha	ISO646-DE	/usr/man/de_DE646de
Alemão	Alemanha	ISO6937	/usr/man/de_DE6937
Alemão	Alemanha	ISO8859-1	/usr/man/de_DE.88591
Alemão	Suiça	ISO646-CH	/usr/man/de_CH.646ch
Japonês	Japão	JIS	/usr/man/ja_JP.jis
Japonês	Japão	SJCS	/usr/man/ja_JP.sjis
Japonês	Japão	UJ (ó EUC-J)	/usr/man/ja_JP.ujis

As páginas do manual para os comandos e informação que se encontra abaixo /usr/local estão armazenadas em **/usr/local/man**. As páginas do manual para o sistema **X Windows** estão armazenadas em **/usr/X11R6/man**. Logo todas as hierarquias de páginas do manual no sistema devem ter a mesma estrutura que /usr/man. Os diretórios vazios podem ser omitidos da hierarquia de páginas do manual. Por exemplo se, /usr/local/man não tem páginas do manual na seção 4 (dispositivos) então se pode omitir /usr/local/man/man4.

As seções da páginas cat (cat[1-9]) que contém **páginas do manual formatadas**, também se encontram dentro os subdiretórios /<mandir>/<locais>, pois não são requeridas nem devem ser distribuidas no lugar das fonte **nroff** das páginas do manual.

As páginas do Manual do sistema de manipulação de e-mail **mh** devem ter o prefixo **mh** em todos os nomes de arquivos das páginas.

As páginas do sistema **X Windows** devem de ter o prefixo **X** em todos os nomes dos arquivos das páginas.

A prática de colocar as páginas do manual de diferentes idiomas, nos subdiretórios aprópriados de /usr/man também se aplica a as outras hierarquias de páginas do manual, tais como /usr/local/man e /usr/X11R6/man. Isto também é aplicável a estrututura opcional de /var/catman, mostrada no subdiretório /var.

5.4.3.12.10 - Subdiretório /usr/sbin

Este diretório contém quaisquer arquivo não essenciais utilizado exclusivamente pelo administrador do sistema.

Os programas de administração do sistema que sejam utilizados para a reparação do sistema, montado no /usr, outras funções essenciais devem localizar-se em /sbin em vez de /usr/bin.

Tipicamente /usr/sbin contém os **deamons** de rede, quaisquer ferramenta de administração não essenciais e arquivos para programas servidores não-críticos. Estes inclue os **deamons** da internet que são chamados por inted (chamados in.*) tais como **in.telnetd** e **in.fingerd** e os deamons basados em **rpc** manipulados por **portmap** (chamados rcp.*), tais como **rcp.infsd e rcp.mountd**.

Estes programas servidores são utilizados quandoocorre um estado que o System V conhece como "run levO2" (estado multi-usuário) e o "run levO3" (estado em rede) ou estado que o BSD conhece como "modo multi-usuário", neste ponto ficam disponíveis os serviços para os usuários (suporte de impressão) e até outras máquinas (por exemplo, exportar NFS).

Os programas administrativos instalados localmente devem estar localizados em: /usr/local/sbin.

5.4.3.12.11 - Subdiretório /usr/share

São informação que independente da arquitetura, quaisquer especificação para /usr/share será incluida em um documento suplementar ao FSSTND, de acordo com a Linux Organization, os pesquisadores do FSSTND acham que /usr/share não é necessário na maioria dos sistemas Linux.

5.4.3.12.12 - Subdiretório /usr/src

Contém o Código fonte para o kern do Linux, qualquer código fonte não local deve localizar-se neste diretório . O único código fonte que sempre deve localizar-se em um lugar específicos é o código do kern(quando exista ou esteja enlaçado como parte de uma estrutura /usr/include). Podem-se usar subdiretórios que desejar.

O código fonte para Kern deve sempre estar em seu lugar mesmos. Os arquivos include do código do kernel. Esses arquivos estão localizados neste diretórios.

/usr/src/linux/include/asm-<arch>
/usr/src/linux/include/linux

/usr/include deve conter links a estes diretórios, chamados asm e linux, dados que são necessitados pelo compilador de C, ao menos estes arquivos include devem sempre ser distribuidos nas intalações que incluem um compilador C. Devem ser distribuidos no diretório/usr/src/linux de forma que não existão problemas quanto os administradores do sistema atualizem sua versão do kern pela primeira vez.

/usr/src/linux pode também ser um links simbólico a um árvore de código fonte do kernel.

5.4.3.13 - A Hierárquia /var

/var	Informação variável
adm	Informações administrativa do sistema (obsoleto). Link simbólico até /var/og
catman	Páginas do manual formatadas localmente
lib	Informação do estado das aplicações
local	Informação variável do software de /usr/local
ock	arquivos de bloqueio
og	arquivos de Agenda
named	arquivos DNS, somente rede
nis	arquivos base de dados NIS
run	arquivos relevantes a processos execução do sistema
spool	Diretórios de trabalhos em fila para realizar-se depois
tmp	arquivos temporários, utilizado para manter /tmp menor possível

/var contém arquivos com informação variável. Esta incluem arquivos e diretórios em fila de execução, informação de ordem administrativa e arquivos temporários e transitorios.
Algumas porção de /var são não compatível entre diferentes sistemas. Por exemplo,

/var/og, /var/ock e /var/run. Outras porção são compatível, notoriamente /var/spool/mail e /var/spool/news.

/var é especificada aqui para fazer possível montar /usr somente para leitura. Tudo aquilo que alguma vez ficou em /usr é escrito durante a operação normal do sistema (não durante a instalação e manutenção do software) deve ir em /var.

Se /var não pode ser uma participação separada, é preferivel mover /var para fora do diretório raiz, pois dentro da partição /usr (Isto se fazia algumas vezes para reduzir o tamanho da partição raiz ou quando há pouco espaço na partição raiz). Sendo assim, /var não deve ser enlaçada a /usr, porque fazia que a separação entre /usr e /var seja mais difícil e seguramente criará um conflito do nomes, em vez links /var e / usr/var.

5.4.3.13.1 - /var/adm : Agenda do sistema e arquivos contabilizados (obsoleto)

Este diretório tem sido repassado para /var/og e outros diretórios . Deve ser um links simbólico a /var/og até que todos os programas não se refiram mas a algum arquivo em /var/adm.

utmp seja movido a /var/run. Todos os arquivos agendas vão ser movidos a /var/og incluindo o arquivo wtmp.

O suporte de empacotamento das distribuições deve armazenar em /var/lib/<nome>.

Nota: O links simbólico /var/adm não deve ser necessário a maioria dos sistemas Linux-i386ELF dado que Otroca foi
introducido antes que ELF fora liberado al público.

5.4.3.13.2 - /var/catman : Páginas do Manual Formatadas localmente (opcional)

Este diretório poporcionara uma localização padrão para os computadores que utilizam uma partição /usr somente para leitura , pois desejam permitir o armazenamento temporário de páginas do manual formateados localmente. Os administradores que montaram /usr como escrita (intalações mono-usuários) podem escolher não usar /var/catman e exibir as páginas do manual formatadas dentro dos diretórios cat[1-9] dentro /usr diretamente. Recomendamos que a maioria dos administradores utilizem uma das seguintes opções em seu lugar.

Preformatando todas as páginas do manual dentro /usr com o programa (catman).

Não será permido o armazenamento temporário das páginas formatadas do manual e precise que se execute **nroff** cada vez que necessite uma página.

Se permita o armazenamento temporário local das páginas do manual em /var/catman.

A estrutura de /var/catman necessita refrear ambos, O trecho da existencia de multiplas hierarquias de página do manual e a possibilidade do uso de multiplos idiomas.

Dada uma página do manual sem formatar que normalmente aparece em /usr/<rota1>/man/man[1-9], a versão formatada armazenada temporáriamente deve ir em /var/catman/<rota2>/cat[1-9], aonde <rota2> é <rota1>. Os componentes <rota2> e <rota1> estão ausente no caso de /usr/man e /var/catman.

Por exemplo, /usr/man/man1/ls.1 é formatado em /var/catman/cat1/ls.1 e /usr/X11R6/man/<locais>/man3/XtCass.3x esta formatado em /var/catman/X11R6/ <locais>/cat3/XtCass.3x .

As páginas do manual escritas em /var/catman/cat[1-9] podem eventualmente, transferir-se a /usr/<rota>/cat[1-9]. De igual forma as páginas do manual formatadas dentro de /usr/<rota>/cat[1-9] podem expirar se não são acceitas num período do

tempo.

Se tivessem páginas do manual preformatadas com um sistema Linux num meio somente para leitura (por exenplo um CD-ROM), devem estar instaladas em /usr/<rota>/cat[1-9]. /var/catman está reenviado para um lugar de armazenamento temporário para páginas de manual formatados.

5.4.3.13.3 - /var/lib : Informação de Estado das Aplicações.

/var/lib.- Informação de Estado das Aplicações

emacs Diretório do estado do Emacs
games Informação variável de jogos
news Arquivos variáveis de Cnews/INN
texmf Informação variável asociada com TeX
xdm arquivos de autenticação e código de erros de aplicações X Windows.

/var/lib/<nome> é o lugar apropriado para o suporte de empacotamento de todas as distribuições. Diferentes distribuições de Linux podem utilizar diferentes nomes para suporte.

/var/lib/emacs

O diretório do estado GNU Emacs, O lugar do **ndos** arquivos de informação independente da arquitetura, que Emacs modifica quando executa, deve ser /var/lib. No presente, Emacs somente localiza seu diretório de arquivos de bloqueio abaixo do diretório de estado (em <direstado >/emacs/lock), pois pode fazer uso mais extenso do mesmo no futuro, notoriamente, só requer a adição de uma opção sensível no programa configure de Emacs para fazer esta troca (antes de compilar).

/var/lib/games

Assim como os subdiretórios antes citados, quaisquer informação variável relacionada com os jogos que se encontram em /usr/games, devem estar aqui. /var/lib/games deve incluir a informação variável que previamente será encontrada em /usr/lib/games. A informação estática, tal como textos de ajuda, descrições do niíveis devem permanecer em /usr/lib/games.

/var/lib/news

/var/lib/news deve usar para armazenar toda a informação variável asociada com os servidores de news tais como Cnews e INN, inclusive o arquivo histórico, o arquivo ativo.

/var/lib/texmf

/var/lib/texmf deve usar para armazenar a informação variável associada com TeX. Particularmente, em /var/lib/texmf/fonts armazenaram todas as fonte tipográficas que são geradas automáticamente por **MakeTeXPK.**

Deve haver um links desde /usr/lib/texmf/fonts/tmp até /usr/lib/texmf/fonts. Este links permite os usuários fazer uso de uma só rota /usr/lib/texmf/fonts/tfm quando houver trocas da sua variável TEXFONTS (Esta é A rota default nas ferramentas TeX de Karl Berre distribuidas por ftp.cs.umb.edu:pub/tex [A razão de

mencionar-lo aqui é que são o padrão de fato nas intalações UNIX, estas ferramentas são amplamente usadas na comunidade LINUX]. Se se utiliza outra distribução de TeX, deve fazer um links deste diretório de fonte apropriada até /usr/lib/texmf/fonts).

O **MakeTeXPK** que se distribue e com **dvipsk** colocará os arquivos .pk em fonts/pk/<dispositivo>/<nome_da_fonte>, (por exemplo, fonts/pk/Canon_CX/cmr10.300pk). Os arquivos .pk podem ser plugados periodicamente do árvore /var/lib/texmf ou pode-se mover dentro da árvore /usr/lib/texmf. Se usarem geradores automáticos de **.mf** ou **.tfm**, estes devem por sua informações nos subdiretórios **mf** ou **tfm** de /var/lib/texmf/fonts.

/var/lib/xdm

/var/lib/xdm contém a informação variável de xdm que consiste nos arquivos xdm-errors e quaisquer arquivo pertencentes a xdm. Os arquivos de xdm tais como **chooser** devem até estar localizados na localidade histórica em /usr/X11R6/lib/X11/xdm. O arquivo xdm-pid devem estar em /var/lib/xdm apesar de existir /var/run. Os arquivos restantes devem estar em /etc/X11/xdm.

5.4.3.13.4 - /var/local : Informação variável do software que está em /usr/local

Este diretório contém toda a informação variável que esta relacionada com o software que se encontra em /usr/local. Naturalmente a implementação desta subdiretório é prerrogativa do administrador do sistema . Como a informação pode estar noutro lugar do diretório/var, não deve colocar em /var/local. Por exemplo, todos os arquivos de bloqueios estarão em /var/ock.

5.4.3.13.5 - /var/ock : arquivos de Bloqueio

Os arquivos de bloqueio devem de armazenar-se dentro uma estrutura do diretório de /var/ock.

Para preservar a habilidade de montar /usr somente para leitura , não se deverá colocar os arquivos de bloqueio na partição /usr.

Os arquivos de boqueio dos dispositivo, tais como os arquivos de boqueio do dispositivos serie que antes se encontravam em /usr/spool/ock ou em /usr/spool/uucp devem agora ser armazenado em /var/ock. A convenção para a nomenclatura que deve utilizar-se é LCK., seguido do nome base do dispositivo. Por exemplo, para bloquear /dev/cua0 se deverá criar o arquivoLCK..cua0.

O formato usado para os arquivos de bloqueios de dispositivo no Linux deverá ser o Formatos arquivos de bloqueio HDB UUCP. O formato HDB é armazenado em OPID (Identificador de processo) com um número decimal na ASCII de 10 bytes, com um caracter de linha nova.

Por exemplo, se o processo 1230 retém um arquivo de bloqueio, contém dados seguinte onze(11) caracteres: espaço, espaço, espaço, espaço, espaço, espaço, um, dois, três, qautro e nova linha.

Então quaisquer coisa que usar /dev/cua0, pode ler o arquivo de bloqueio e atuar de acordo (todos os arquivos de bloqueio em /var/ock devem ser lidos por todos).

54.3.13.6 - /var/og : Arquivos agenda e diretórios

Este diretório contém arquivos agenda misceláneos. A maioria dos arquivos agenda se devem exibir neste diretórios ou subdiretórios aprópriados.

astog Registro do último acesso de cada usuário
mesageMensagem do sistema desde que logou ao sistema
wtmp Registro de todos os acessos e saídas

Pode requerer um links simbólico desde /var/og/utmp até /var/run/utmp basta que nenhum programa se refira a /var/adm/utmp (/var/adm é em si mesmo um links simbólico transicional até /var/og).

5.4.3.13.7 - /var/named : arquivos DNS

Este diretório contém todos os arquivos de trabalho do servidor de nomes Internet, named. Recomendamos que /etc/named.boot seja um links simbólico até /var/named/named.boot, dado que /etc/named.boot é o arquivo de inicialização default, se não fornece argumentos a named.

5.4.3.13.8 - /var/nis

Arquivos de base de dados do serviço de informação de rede (NIS) o sistema de informação de rede (NIS) era anteriormente conhecido como as páginas Amarelas Sun. A funcionalidade e localização de diretórios de ambos é o mesmo pois o nome (yelow Page) é uma marca registrada no Reino Unido , pertencem a Bristish Telecommunications plc. e não pode ser usada sem permisão.

5.4.3.13.9 - /var/preview:arquivos guardados depois de uma colisão ou uma termino inesperado, exemplo vi (editor de texto).

Este diretório contém os arquivos que são armazenados antes de quaisquer terminação não esperada de ex. vi.

5.4.3.13.10 - /var/run : arquivos variáveis de tempo de execução

Este diretório contém arquivos com informação do sistema que o descrevem desde que inicializou. Geralmente os arquivos neste diretório devem ser deletar (remover ou truncar) aol començar o processo de inicialização.

Os arquivos identificados do processo (PID), que estavam originalmente /etc, devem colocar em /var/run. A convenção de nomenclatura dos arquivos PID é <nome-programa>.pid, por exemplo o arquivo PID de crond se chama /var/run/crond.pid.

O formato interno dos arquivos PID permanecem sem troca. O arquivo deve conter o indicador de ponto decimal codificado como ASCII, seguido por um caracter nova linha. Por exemplo, o processo número 25, /var/run/cond.pid conterá 3 caracteres, dos cinco e nova linha.

Os programas que leiam arquivos PID devem ser flexível na acepção, por exemplo devem ignorar os espaços extras, zeros a esquerda, ausência do caracter nova linha ou linhas adicionais no arquivo PID. Os programas que criam arquivos PID devem utilizar a especificação dada no parágrafo anterior.

O arquivo utmp, que armazena informação acerca de quem está atualmente utilizando o sistema, se localiza neste subdiretório .

Os programas que mantenham sockets transitorios de dominio UNIX, devem colocaá-los neste diretório.

5.4.3.13.11 - /var/spool

Diretórios de fila de trabalhos para procedimento posterior /var/spool é tradicionalmente utilizado para a informação local de máquina que é enviada para processo depois, até o subsistemas UNIX. Por exemplo, trabalhos de impressão que são armazenados aqui para entrega posterior ao daemon da impressora, o e-mail que saí é armazenado aqui para entrega a sistemas remotos e os arquivos UUCP são armazenados aqui para transmissão dos sistemas UUCP vencidos o e-mail que entra e as noticias são armazenadas aqui para entregar-se aos usuários e os trabalhos de at e cron são armazenados aqui para execução posterior pelo daemon cron.

/var/spool

at Trabalhos de at

cron Trabalhos de cron
lpd Diretório de impressora *
mail arquivos caixa-postal dos usuários
mqueueFila de espera dos correio
news Diretório de noticias *
rwhod arquivos rwhod
smail Diretório de smail *
uucp Diretório de UUCP
Nota:
* Segnifica fila de trabalhos para processamento posterior.

Os arquivos de bloqueio UUCP devem localizar-se em /var/ock. Veja a seção acerca de /var/ock.

/var/spool/lpd

/var/spool/lpd --- Diretório de fila de trabalhos para processamento posterior a impressão
<impressora> --- Diretório que tem a fila especifica desta impressora

O arquivo de bloqueio para lpd, lpd.ock deve estar localizado em /var/spool/lpd. O arquivo de bloqueios de cada impressora deve localizar-se no diretório<impressora> da impressora especifica e deve chamar ock.

5.4.3.13.12 - /var/tmp : Arquivos temporários, utilizando para manter /tmp pequeno.

Os arquivos que estão no /var/tmp estão armazenados por uma duração não especifica. (Lembre-se que os diretórios temporários do sistema não garantiram manter a informação por nenhum período particular).

A informação armazenada em /var/tmp tipicamente esta numa "forma definida localmente", pois usualmente é menos frequentemente que /tmp. Se pode encontar informação sobre diretórios temporários na seleção dedicada a /tmp (acima).

Deve existir um links simbólico desde /usr/tmp até var/tmp por razão de compatibilidade.

5.4.4 - Alguns dilemas sobre o Sistema de Arquivos

A rede apresenta um dilema intirerante, algumas pessoas quiseram separar os arquivos de rede e configuração dos outros arquivos de configuração. Ou seja, estão em desacordo . Sentimos que a rede não é um "pacote", senão uma parte integral da maioria das máquinas UNIX (e similares). Não se deve colocar a rede em um só diretório senão localizar-se sistematicamente nos diretórios aprópriados.

/bin **{hostname, netstat, ping}** Qualquer coisa que algum usuário queiram utilizar considerado vital.
/sbin **{arp, ifconfig, route}** Qualquer coisa que só root necessita e considera vital.
/usr/bin **{finger, rep, rogin, telnet, etc.}** Alguns arquivos que algum usuário

queira utilizar e que não são vitais.
/usr/sbin **{in.ftpd, inetd, lpd, portmap, etc..}** Alguns arquivos que somente o administrador utiliza, que não são vitais.

Pode parecer confuso a princípio (leva tempo digerindo), tem sentido . Se por alguma razão você só pode montar a partição raiz, e necessita acessar a rede para reparar seu sistema, não quer que os arquivos estão em /usr/etc (como estão algumas vezes). Os arquivos que necessitam para montar /usr as situações normais (e de emergência) estão coocados dentro da sub-árvore raiz, e quaisquer outros podem colocar em /usr, para manter O tamanho do sistema de arquivos raiz pequeno.

Os arquivos de configuração para a rede pertencem a /etc.

A Estruturas independente da arquitetura, o diretório/usr/share tipicamente contém arquivos independente da arquitetura, tais como páginas do manual, fuso horário, informação de terminais, etc. No momento presente não há diferentes arquiteturas para Linux, pois com o tempo, veremos que Linux incluirá outras arquiteturas e outros sistemas similares a UNIX.

Nota: Nenhum programa nunca deverá fazer referência a alguma coisa em /usr/share. Por exemplo, um programa de páginas do manual não deve nunca buscar diretamente /usr/share/man/man1/1s.1, sempre deve referir a /usr/man/man1/1s.1. Qualquer coisa em /usr/share, será "apontada" através do uso do enlacé símbolos de outras áreas do sistema de arquivos, tais como /usr/man, /usr/lib/<algo>, etc.

Até se trabalhar as especificações de /usr/share.

Os Link simbólicos, existe muitos usos para os link simbólicos em cada sistemas de arquivos. Embora este documento como esta não respalda o uso do link simbólicos na implementação default (os encontrados depois de instalar Linux), usam frequentemente com bons propósitos em diferentes sistemas. O ponto é que os link simbólicos devem estar ali para manter todos os arquivos e diretórios onde cada usuário espera encontar.

Estar preparado para acertar que certos diretórios, até aqueles contidos no diretório raiz, até sejam link simbólicos. Por exemplo em alguns sistemas /home não estará na raiz, senão enlaçado simbolicamente a um diretório/var ou algum outro lugar. /home poderia ter também sua própria partição física e desde logo, ser montada como tal.

Similarmente, dado que /usr poderia estar em um servidor de arquivos central montado vía NFS, /usr/local pode-se enlaçar simbolicamente a /var/local. Este troca pode-se justificar recordando A razão principal de ter /var: separadas de diretórios de arquivos que variam com o tempo e entre diferentes sistemas e máquinas daqueles que podem compartilhar e sejam somente para leitura .

Alguns sistemas além disso enlaçar /tmp a /var/<algo> se a partição raiz se torne muito pequena (ou é muito pequena). Existe mais exemplos de bons usos de link simbólicos, pois todo o assunto não se reduz a estas coisas: os pacotes devem ser capazes de encontar as coisas onde esperam (razoavelmente) e os link simbólicos pode-se utilizar para resolver os problemas de muitos casos. Ou seja, se podem gerar problemas com o uso demasimados link simbólicos. Este problema influi sobre a confiança nos link simbólicos para resolver problemas, confusão resultante do sobre o uso do link simbólicos e as preferências estéticas das diferentes pessoas.

Os Arquivos Compilados Estáticamente o Linux executa atualmente em uma gama de sistemas, alguns com somente um usuário e disco pequeno, outros como servidores em ambiente com rede muito grande, dada esta variedade , esta documento não impõe regra sobre quais arquivos estão compilados estaticamente ou dinamicamente , com as seguinte exceções. Ambos **ln** e se **nc**, devem existir em /bin; quaisquer versão estática pode-se colocar em /sbin ou repassa-la em /bin.

Os grande sistemas Linux podem desejar incluir outros arquivos **estáticos (sh,**

init, mkfs, fsch, tunefs, mount, umount, swapon, swopff, gette , login e outros). Os instaladores e os administradores de sistemas, são livres para conectar dinâmicamente ou estaticamente estes outros arquivos segundo sua conveniência, sempre que a localização dos arquivos arquivos não troque.

Em sistemas de rede, (especialmente aqueles que não tem unidade de disco flexivel), podem querer compilar estaticamente **ifconfig**, **route, hostname** e outras ferramentas de rede. Isto usualmente não é necessário.

5.4.5 - Descrições sucinta do conteúdo dos manuais.

man1: Programas de usuário.

As páginas deste manual descrevem os comandos aceitos publicamente. A maioria da docomentação dos programas que o usuário necessita encontra-se aqui.

man2: Chamadas do Sistema.

Esta seção descreve todas as chamadas do sistema (requisição do kern do Linux para realizar certas operações).

man3: Subrotinas e funções de bibliloteca.

Descreve programas e rotinas da bibliloteca que não são chamadas diretas do serviços do kernel. Esta seção e a **man2** são de interessante de programdores.

man4: arquivos especiais.

Esta seção descreve os arquivos especiais, funções relacionadas com os manipuladores e o suporte a rede que estão disponíveis no sistema. Esta documentação incluem os arquivos de dispositivo que encontram-se em /dev e a interface do kern para suporte de protocolos de rede.

man5: Formatos de arquivos.

Aqui se encontram os formatos para muitos dos arquivos cujo o formato não seja intuitivo. Esta incluem vários arquivos include, arquivos de saída de programas, e arquivos de sistema.

man6: Jogos.(arquivos educativos)

Esta seção do comenta os jogos, demos e programas triviais.

man7: Miscelânia

As páginas do manual que são difícil de classificar e designar, pertencente a seção 7. As **troff** e outros macro pacotes de procedimento do texto encontram-se aqui.

man8: Administração do Sistema

Aqui esta a documentação dos programas utilizados pelos administradores de sistemas para a operação e manutenção. Alguns destes programas são ocasionalmente utilitários para usuários normais.

man9: Funções e variáveis internas do kernel

Contém informações do código fonte do kern no Sistemas Linux.

6 - Pontos Positivos e negativos

O Linux é sem dúvida a melhor opção de Unix para PC, pois possui todas as características um UNIX moderno, tais como : multitarefa real, multiusuário, memória virtual, biblioteca compartilhada, interface gráfica (X Windows) etc. O Linux possui centenas de comandos embutidos, chamado utilitários e ferramentas, cada ferramenta é um programa distinto, destinado a fazer um tarefa específica de forma rápida e segura, vide item 5.4.3 - Compisição dos diretórios do Linux.

O fato de ser um sistema aberto é extremamente flexível (usuário tem acesso ao fonte do sistema) é outro ponto positivo.

O preço é outro atrativo US$ 44,00 (no Brasil, US$ 22,00 nos E.U.A), incluem 6 CD's do Linux Developer's Resourses CD-ROM, distribuido pela InfoMagic.

Diversos grupos de estudo do Linux no mundo inteiro garante atualizações do software praticamente mensais. Aliado a isto, cada nova versão incorpora dispositivos períféricos que são lançado no mercado, trazendo a seu usuário suporte as mais recentes conquista da indústria do hardware.

O Linux tem excelente mercado a nível acadêmico, o que nos faz crer constantes melhoras no software, pois grande parte dos melhores professores/pesquisadores de sistemas operacionais colaboram para o seu desenvolvimneto tecnológico.

A documentação é detalhada e completa (em inglês), em português é bastante escassa (ponto negativo). Toda a documentação pode ser facilmente acessada pela internet em diversos sites ou na documentação que acompanha o software.

O suporte técnico é um dos pontos fracos do sistema, é feito basicamente através da internet, não existe nenhuma empresa no Brasil especializada no suporte ao Linux.

O número de aplicativos é limitado e não existe a curto prazo pespectiva de entrada de grandes software house desenvolvendo aplicativos para Linux, sem aplicativos não há como um sistema se tornar popular no mundo dos PCs.

O sistema de arquivos varia de um distribuição a outro, tornado difícil a vida do administrador de sistema, que muitas vezes tem dificuldade de descobrir o que essencial em cada subdiretório, limpar, criar, preparar, verificar, encontar e montar outros sistemas de arquivos (possívelmente em máquinas remotas), todas estas tarefas podem ser dificultadas se não encontrarmos os arquivos aonde esperamos.

A administração e a operação de um modo geral é bem mais complexa em um ambiente unix, inclusive o Linux, do que no ambiente DOS/Windows, o que dificulta sua popularização.

A Interface Gráfica X-Windows, Xfree86 versão 3.2, ainda precisa ser melhorada, principalmente os aplicativos que são desenvolvidos para a mesma, ainda muito pobres comparada com a GUI do Windows95.

Conclusão

O Linux, é um sistema operacional do tipo Unix, o padrão System V esta embutido no seu kernel, foi desenvolvido para a plataforma IBM-PC, sendo assim, ele possui a robustez e segurança e flexibilidade do Unix. Além disso ele possui uma interface gráfica chamada de X Windows(XFree86 versão 3.2, por exemplo, existe outrasinterface gráfica), que é semelhante a do Windows 95, menos sofisticada, menos aplicativos e menos elaborada, porém funcional.

O Projeto Linux foi desenvolvido para ser uma arquitetura aberta, você terá toda a liberdade de desenvolver software para sua plataforma. Os fontes são distribuidos junto com o produto. Por outro lado, não existe um grande mercado para a plataforma Linux, sendo assim, as grandes softwares houses do mundo, como a Microsoft, Lotus, Corel, Borland, Novell, etc não se preocuparam em desenvolvem aplicativos para ele. O desenvolvimento de aplicativos ainda é pequeno, colaboradores, em sua maioria pesquisadores, desenvolve os softwares e a distribuição é feita, preferencialmente, no mesmo pacote a preços módicos, todo o pacote custa no Brasil US$ 44,00 (US$ 22,00 nos USA), pode ser comprado utilizando qualquer Cartão de Crédito Internacional.

Outro problema do Linux é a falta de suporte técnico, não existe a nível comercial no Brasil, as informações são obtidas através de diversos manuais contidos no software, ou através da internet. Existe uma farta documentação disponível na rede, diversos servidores WWW (World Wide Web), lista de discussões (serviços que permite o intercâmbio de mensagem entre vários usuários, funciona como uma extensão do correio eletrônico, onde qualquer mensagem enviada a este endereço fictício, conhecido como alias, e reenviada automaticamente para todos os endereços da lista associada), Netnews ou USENET ou NEWS (semelhante a lista de discussão, só que as mensagem são enviadas a um determinado computador da rede que as reenvia em bloco, para outros computadores que aceitam o serviço), FTP (File Transfer Protocol, serviço básico de tranferência de arquivos na rede), etc.

O mercado do Linux no Brasil é restrito, praticamente, ao meio acadêmico e alguns provedores da Internet (ex.: Universidade Federal de Goiás utiliza como roteador e servidor de WWW em algumas unidade acadêmicas). É difícil acreditar no crescimento do mercado Linux no Brasil a curto e a médio prazo fora deste nincho de mercado, devido a sua dificuldade de operação, o Linux foi projetado por programadores para programadores, a fim de ser utilizado em ambiente onde a maioria dos usuários tenha uma certa experiência, soma-se a isto a falta de software para a plataforma Linux que dificulta a sua popularização. As empresas que utilizam o Unix comercialmente na plataforma RISC, utilizam sistemas proprietários desenvolvidos por empresas como Sun, IBM, Dec, etc, que fazem tanto o hardware quanto SO, embora exista versões do Linux para algumas destas plataformas (ex. Sun) não acreditamos no crescimento do Linux neste mercado, pois é estritamente fechado.

Apesar disto, o Linux é uma opção séria como sistemas operacional do tipo Unix para o mundo PC, podemos recomendá-lo sem medo de errar, se sua empresa ou aplicação precisar de um sistema com as características do Unix, é você possue máquinas Intel ou compatíveis, pode utillizá-lo é a melhor opção hoje e com grandes possibilidades de crescimento nesta faixa de mercado.

Apêndices

A - Comandos básicos do sistema UNIX

Principais comandos - ordem albabética

Comandos em Unix possuem algumas características particulares. Eles podem ser controlados por opções e devem ser digitados em letras minúsculas.

1 - cat : Oficialmente usado para concatenar arquivos. Também usado para exibir todo o conteúdo de um arquivo de uma só vez, sem pausa.

Sintaxe: cat < arquivo1 > < arquivo2 >... < arquivo n >,

onde (arquivo1) até (arquivo n) são os arquivos a serem mostrados. "cat" lê cada arquivo em sequencia e exibe-o na saída padrão. Deste modo , a linha de comando:

cat < arquivo >

exibirá o arquivo em seu terminal; e a linha de comando :

cat < arquivo1 > < arquivo2 > > < arquivo3 >

concatenará "arquivo1" e "arquivo2", e escreverá o resultado em . O símbolo ">", usado para redirecionar a saída para um arquivo, tem caráter destrutivo; em outras palavras, o comando acima escreverá por cima do conteúdo de < arquivo3 >. Se, ao invés disto, você redirecionar com o símbolo ">>", a saída será adicionada a <arquivo3 >, ao invés de escrever por cima de seu conteúdo.

2 - cd : Muda o diretório de trabalho corrente.

Sintaxe : cd < diretório >

onde (diretório) é o nome do diretório para o qual você deseja mudar. O símbolo "." refere-se ao diretório corrente e o símbolo ".." refere-se ao "diretório-pai". Para mover para um "diretório-pai", ou seja, um diretório acima do que você está, use o comando :

cd ..

(Note o espaço entre "cd" e "..")

Você também pode usar nomes-de-caminho (pathnames) como argumento para o comando cd. Por exemplo :

cd /diretorio1/diretorio2

o posicionará diretamente em "diretório2". O uso de "cd" sem nenhum argumento fará com que você retorne para o seu "home-directory" .

3 - chgrp : Modifica o grupo de um arquivo ou diretório.

Sintaxe: chgrp [-f] [-h] [-R] gid nome-do-arquivo

"chgrp" modifica o identificador de grupo ("group ID" , gid) dos arquivos passados como argumentos.
"gid" pode ser um número decimal especificando o group id, ou um nome de grupo encontrado no arquivo "/etc/group". Você deve ser o proprietário do arquivo, ou o super-usuário, para que possa utilizar este comando.

Opções
Saída
-f
Esta opção não reporta erros
-h
Se o arquivo for um link simbólico, esta opção modifica o grupo do link simbólico. Sem esta opção, o grupo do arquivo referenciado pelo link simbólico é modificado.
- R
Esta opção é recursiva."chgrp" percorre o diretório e os subdiretórios, modificando o GID à medida em que prossegue.

4 - chmod : Modifica as permissões de um arquivo ou diretório.Você deve ser o proprietário de um arquivo ou diretório, ou ter acesso ao root, para modificar as suas permissões.

Sintaxe : chmod permissões nome_do_arquivo

onde :

permissões - indica as permissões a serem modificadas;
nome - indica o nome do arquivo ou diretório cujas permissões serão afetadas.

As permissões podem ser especificadas de várias maneiras. Aqui está uma das formas mais simples de realizarmos esta operação :

1- Use uma ou mais letras indicando os usuários envolvidos:

. u (para o usuário)
. g (para o grupo)
. o (para "outros")
. a (para todas as categorias acima)

2- Indique se as permissões serão adicionadas (+) ou removidas (-).

3- Use uma ou mais letras indicando as permissões envolvidas :

. r (para "read") (ler)
. w (para "write") (escrever)
. x (para "execute") (executar)

Exemplo : No exemplo a seguir, a permissão de escrita ("write") é adicionada ao diretório "dir1" para usuários pertencentes ao mesmo grupo. (Portanto, o argumento "permissões" é g+w e o argumento "nome" é dir1).

```
$ ls -l dir1
drwxr-xr-x 3 dir1      1024 Feb 10 11:15 dir1
$ chmod g+w dir1
$ ls -l dir1
drwxrwxr-x  3 dir1     1024 Feb 10 11:17 dir1
$
```

Como você pôde verificar, o hífen (-) no conjunto de caracteres para grupo foi modificado para "w" como resultado deste comando.

Quando você cria um novo arquivo ou diretório, o sistema associa permissões automaticamente. Geralmente, a configuração "default" (assumida) para os novos arquivos é:

- r w - r - - r - -

e para novos diretórios é:

d r w x r - x r - x

5 - chown : Modifica o proprietário de um arquivo ou diretório.

Sintaxe: chown [-fhR] (proprietário) (nome-do-arquivo)

O argumento "proprietário" especifica o novo proprietário do arquivo.Este argumento deve ser ou um número decimal especificando o userid do usuário ou um "login name" encontrado no arquivo "/etc/passwd".

Somente o proprietário do arquivo (ou o super-usuário) pode modificar o proprietário deste arquivo.

Opcões

Saída

- f

Esta opção não reporta erros.

- h

Se o arquivo for um link simbólico, esta opção modifica o proprietário do link simbólico. Sem esta opção, o proprietário do arquivo referenciado pelo link simbólico é modificado.

- R

Esta opção é recursiva."chown" percorre o diretório e os subdiretórios, modificando as propriedades à medida em que prossegue.

6 - cp : Copia arquivos para um outro arquivo ou diretório.

Sintaxe: cp (arquivo1) (arquivo2) ... (arquivo n) (destino)

onde (arquivo1) até (arquivo n) são os arquivos a serem copiados, e (destino) é o arquivo ou o diretório para onde os arquivos serão copiados. O(s) arquivo(s) fonte(s) e o (destino) não podem ter o mesmo nome. Se o arquivo-destino não existe, "cp" criará um arquivo com o nome especificado em . Se o arquivo-destino já existia antes e nào for um diretório, "cp" escreverá o novo conteúdo por cima do antigo.

Exemplo : $ cp -r temp temp1

Este comando copia todos os arquivos e subdiretórios dentro do diretório temp para um novo diretório temp1. Esta é uma cópia recursiva, como designado pela opcão -r. Se você tentar copiar um diretório sem utilizar esta opcão, você verá uma mensagem de erro.

7 - du : Exibe o espaço ocupado de um diretório e de todos os seus subdiretórios, em blocos de 512 bytes; isto é, unidades de 512 bytes ou caracteres..

"du" mostra a utilização do disco em cada subdiretório.

8 - date : Exibe a data configurada no sistema.

O comando "date",a nível de usuário, exibe na tela a data configurada no sistema. Ele pode se usado com opções mostram a data local ou data universal GMT - Greenwich Mean Time. A configuração dos dados deste comando só podem se realizadas pelo super-usuário.

Para exibir a data local, basta executar "date". Caso queira a data GMT utilize a opção "-u".

Veja:

```
%date
Wed Jan 8 12:05:57 EDT 1997
```

Aqui a data é exibida em 6 campos que representam o dia da semana abreviado, o mês do ano abreviado, o dia do mês, a hora disposta em horas/minutos/segundos, a zona horária e o ano.

9 - file : Exibe o tipo de um arquivo.

Alguns arquivos, tais como arquivos binários e executáveis, não podem ser visualizados na tela. O comando "file" pode ser útil se você não tem certeza sobre o tipo do arquivo. O uso do comando permitirá a visualização do tipo do arquivo.

Exemplo :

```
$file copyfile
copyfile: ascii text
```

10 - grep : Exibe todas as linhas, dos arquivos especificados, que contém um certo padrão.

O comando "grep" exibe todas as linhas, dos arquivos nomeados, que são iguais ao padrão especificado.

Sintaxe:

```
grep [padrão] <arquivo_1> <arquivo_2> ... <arquivo_n>
```

onde [padrão] é uma expressão regular, e "arquivo_1" até "arquivo_n" são os arquivos nos quais a procura será feita.

Por exemplo, o comando

grep trabalho /trabalho/unix/grep.html

mostrará todas as linhas no arquivo /trabalho/unix/grep.html que contém o padrão "trabalho".

11 - ls : Exibe informações sobre arquivos nomeados e diretórios, é usado para visualizar o conteúdo de um diretório.

Sintaxe: ls (diretório)[opções]

Quando executado sem qualquer parâmetro, mostra o conteúdo do diretório corrente.Assim, a linha de comando:

$ ls

mostra o conteúdo do diretório corrente naquele momento.Como na maioria dos comandos UNIX, "ls" pode ser controlado por opções que começam com um hífen (-). Tenha sempre o cuidado de deixar um espaço antes do hífen. Uma opção bastante útil é -a (que vem do inglês 'all', tudo), e irá mostrar detalhes que você nunca imaginou sobre o seu diretório. Por exemplo:

```
$ cd
$ ls -a
```

Digitando estes comandos em sequência, o sistema vai para o seu home directory, através do comando cd e em seguida mostra o conteúdo do mesmo, que será exibido da seguinte forma:

```
.              .bacshrc        .fvwmrc
..             .emacs          .xinitrc
.bash_history       .exrc
```

Aqui, o ponto simples refere-se ao diretório corrente, e o ponto duplo refere-se ao diretório imediatamente acima dele. Mas o que são estes outros arquivos que se iniciam com um ponto? Eles são chamados arquivos escondidos. A colocação do ponto na frente de seus nomes os impede de serem mostrados durante um comando "ls" normal.

Outra opção bastante utilizada é -l (que vem do inglês "long"). Ela mosta informação extra sobre os arquivos . Assim, o comando:

$ ls -l

mostra, além do conteúdo do diretório, todas os detalhes sobre cada arquivo pertencente a ele. Por exemplo, suponha que você tenha executado este comando e na tela apareceu algo assim:

```
-rw-r--r--   1  xyz   users   2321   Mar 15 1994    Fontmap
-rw-r--r--   1  xyz   users  14567   Feb  3 1995    file003
```

drwxr-xr-x 2 xyz users 1024 Apr 23 1995 Programs
drwxr-xr-x 3 xyz users 1024 Apr 30 1995 bitmaps

Lendo da esquerda para direita, este primeiro caracter indica se o arquivo é um diretório (d) ou um arquivo comum (-). Em seguida temos as permissões de acesso ao arquivo, sendo as três primeiras referentes ao proprietário, as seguintes ao grupo e, por último, aos demais usuários.

A segunda coluna desta listagem mostra o número de links que o arquivo possui.

A terceira coluna mostra o proprietário do referido arquivo, neste caso, o usuário cujo user name é "xyz".

Na próxima coluna é mostrado o grupo ao qual pertence o proprietário do arquivo (no exemplo temos o grupo users). Na quinta coluna temos o tamanho do arquivo em bytes.

Por fim, na sexta e sétima colunas, temos a data da última modificação feita no arquivo e o nome do mesmo, respectivamente. Vale lembrar que várias opções podem ser usadas de forma composta. Por exemplo, podemos executar o comando:

$ ls -la

e este mostrará todos os detalhes que as opções -l e -a dispõem.

12 - man : Exibe uma página do manual interno do Unix, para um dado comando ou ou recurso (isto é, qualquer utilitário do sistema que não seja comando, por exemplo, uma função de biblioteca). É como um "help" interno ao sistema.

Sintaxe :

man <comando>

onde "comando" e o nome do comando ou recurso que se deseja obter a ajuda.

13 - mkdir : Cria usado para a criação de novos diretórios.

Sintaxe : mkdir (diretório 1) (diretório 2) ...(diretório n)

onde (diretório 1) até (diretório n) são os diretórios a serem criados.

As entradas padrão em um diretório (por exemplo, os arquivos ".", para o próprio diretório, e ".." para o diretório pai) são criadas automaticamente. A criação de um diretório requer permissão de escrita no diretório pai.

O identificador de proprietário (owner id), e o identificador de grupo (group id) dos novos diretórios sáo configurados para os identificadores de proprietário e de grupo do usuário efetivo, respectivamente.

* Opções:

-m (mode)

Esta opção permite aos usuários especificar o modo a ser usado para os novos diretórios.

-p

Com esta opção, mkdir cria o nome do diretório através da criação de todos os diretórios-pai não existentes primeiro.

Exemplo:

mkdir -p diretório 1/diretório 2/diretório 3

cria a estrutura de subdiretórios "diretório 1/diretório 2/diretório 3".

14 - more : Exibe o conteúdo de arquivos nomeados, fazendo pausas a cada tela cheia.

Ele exibe um arquivo, uma tela cheia de cada vez, fazendo normalmente uma pausa após cada tela cheia, quando exibe "--More-- " na parte de baixo da tela.

Ao teclar-se (Enter), more irá exibir uma linha a mais; ele exibe outra tela cheia ao teclar-se o caracter "espaço". O caracter "b" faz com que "more" exiba a tela anterior. O caracter "q" provoca a parada de execução do comando more.

Sintaxe: more (arquivo 1) (arquivo 2) ... (arquivo n)

onde (arquivo 1) até (arquivo n) são os arquivos a serem exibidos.

Pode-se procurar por uma palavra (ou uma cadeia de caracteres) em um arquivo. Para isso, pressione o caracter "/", digite a palavra (ou a cadeia de caracteres),e tecle (Enter).

* Opções:

-c

Limpa a tela antes de exibir o conteúdo do arquivo. Esta opção é ignorada se o terminal não tem a abilidade para limpar até o final de uma linha.

15 - mv : Move arquivos para um outro arquivo ou diretório.

O comando "mv" é utilizado para mover arquivo(s) para outro arquivo ou diretório. Este comando faz o equivalente a uma cópia seguida pela deleção do arquivo original. Pode ser usado para renomear arquivos.

Sintaxe: mv (arquivo 1) (arquivo 2) ... (arquivo n) (destino)

onde (arquivo 1) até (arquivo n) são os arquivos a serem movidos, e (destino) é o arquivo ou o diretório para onde os arquivos serão movidos.

Se (destino) não for um diretório, somente um arquivo deverá ser especificado como fonte. Se for um diretório, mais de um arquivo poderá ser especificado.

Se (destino) não existir, "mv" criará um arquivo com o nome especificado. Se (destino) existir e não for um diretório, seu conteúdo será apagado e o novo conteúdo será escrito no lugar do antigo. Se (destino) for um diretório, o(s) arquivo(s) será(ão) movido(s) para este diretório.

Os arquivos "fonte" e "destino" não precisam compartilhar o mesmo diretório pai.

*Opções:

-i

Com esta opção, "mv" irá perguntar a você se é permitido escrever por cima do conteúdo de um arquivo destino existente.
Uma resposta "y" (yes = sim) significa que a operação poderá ser executada. Qualquer outra resposta impedirá que "mv" escreva por cima do conteúdo de um arquivo já existente.

Exemplo:
```
$ pwd
/home/usuário1/temp
$ ls
teste
$ mv teste ../temp1
$ ls ../temp1
teste
```

Neste exemplo, o diretório teste foi movido de temp para temp1 com o comando "mv".

16 - passwd : Modifica a senha pessoal.

Para garantir a segurança do sistema, o sistema Unix requer o uso de uma senha. Se você achar que alguém utilizou sua conta sem permissão, mude sua senha imediatamente.

Algumas dicas para a escolha da senha:

- Escolha uma senha que você possa lembrar sem a necessidade de escrever.
- A sua password deve ter ao menos seis caracteres e deve conter pelo menos um número.
- Não use seu próprio nome ou suas iniciais.
- Não use nomes de animais ou objetos relacionados a seu interesse.
- Se você tem mais de uma conta, não use a mesma senha para todas as contas.

O comando "passwd" é utilizado para modificar a senha pessoal.

A seguir estão os passos que acontecem quando "passwd" é utilizado:

```
$ passwd
Changing password for (nome-do-usuário)
Old password:
New password:
Retype new password:
$
```

-> Quando o sistema pedir "Old Password:" , digite sua senha atual.

Se nenhuma senha estiver associada a sua conta, o sistema irá omitir este prompt. Note que o sistema não mostra a senha que você digita na tela. Isto previne que outros usuários descubram sua senha.

-> Quando o sistema pedir "New Password:", digite sua nova senha.

-> O último prompt , "Retype new password", pede que você digite a nova senha novamente.

Se você não digitar a senha da mesma maneira em que digitou da primeira vez, o sistema se recusa a modificar a senha e exibe a mensagem "Sorry".

17 - pwd : Exibe o diretório corrente.

Este comando é utilizado para exibir o seu diretório corrente no sistema de arquivos.

18 - rm : Este comando é utilizado para apagar arquivos. É importante lembrar que quando os arquivos são apagados, no sistema Unix, é impossível recuperá-los.

Sintaxe: rm (arquivo 1) (arquivo 2) ... (arquivo n)

onde (arquivo 1) até (arquivo n) são os arquivos a serem apagados.

Se um arquivo não possuir permissão de escrita e a saída-padrão for um terminal, todo o conjunto de permissões do arquivo será exibido, seguido por um ponto de interrogação. É um pedido de confirmação. Se a resposta começar com "y" ("yes" = sim), o arquivo será apagado, caso contrário ele será mantido no sistema.

Quando você apaga um arquivo com o comando "rm", você está apagando somente um link (ligação ou entrada) para um arquivo. Um arquivo somente será apagado verdadeiramente do sistema quando ele não possuir mais nenhuma ligação para ele, isto é, nenhum link referenciando-o. Geralmente, arquivos possuem somente um link, portanto o uso do comando "rm" irá apagar o(s) arquivo(s). No entanto, se um arquivo possuir muitos links, o uso de "rm" irá apagar somente uma ligação; neste caso, para apagar o arquivo, é necessário que você apague todos os links para este arquivo.

Você pode verificar o número de links que um arquivo possui utilizando o comando ls, com a opção "-l".

*Opções:

-f

Remove todos os arquivos (mesmo se estiverem com proteção de escrita) em um diretório sem pedir confirmação do usuário.

-i

Esta opção pedirá uma confirmação do usuário antes de apagar o(s) arquivo(s) especificado(s).

-r

Opcão recursiva para remover um diretório e todo o seu conteúdo, incluindo quaisquer subdiretórios e seus arquivos.

-> CUIDADO : diretórios e seus conteúdos removidos com o comando "rm -r" não podem ser recuperados.

19 - rmdir : é utilizado para apaga diretórios vazios.

Sintaxe: rmdir (diretório 1) (diretório 2) ... (diretório n)

onde (diretório 1) até (diretório n) são os diretórios a serem apagados.

O comando "rmdir" se recusa a apagar um diretório inexistente, exibindo a mensagem:

rmdir : (nome-do-diretório) : No such file or directory

Quando usar "rmdir", lembre-se que o seu diretório de trabalho corrente não pode estar contido no(s) diretório(s) a ser(em) apagado(s). Se você tentar remover seu próprio diretório corrente, será exibida a seguinte mensagem:

rmdir : . : Operation not permited

Se o diretório o qual você deseja remover não estiver vazio, utilize o comando "cd" para acessar os arquivos dentro do diretório, e então remova estes arquivos utilizando o comando "rm".

Opções:

-p

Permite aos usuários remover o diretório e seu diretório pai, o qual se torna vazio. Uma mensagem será exibida na saída padrão informando se o caminho ("path") inteiro foi removido ou se parte do caminho persiste por algum motivo.

-> CUIDADO : diretórios removidos com o comando "rmdir" não podem ser recuperados!

20 - touch : atualiza o último tempo de acesso e/ou modificacão de um arquivo.

Um arquivo vazio será criado se o nome especificado ainda não existir.

Sintaxe: touch [opcões] [mmddhhMM[yy]] nome-do-arquivo

sendo : mm mês
dd dia
hh hora
MM minuto
yy ano (últimos dois dígitos)

Se não for especificada nenhuma data, a data atual será utilizada.

*Opcões:

-a

Atualiza somente o tempo de acesso respectivo.

-m

Atualiza somente o tempo de modificacão.

-c

Previne a criacão de um arquivo se ele não existia anteriormente.

-> As opções default são : -am.

Fonte: Universidade Federal de Goiás, Instituto de Informática, home-page do Projeto de Apoio ao Usuário Internet (http://www.dei.ufg.br/~apoio/unix.html)

B - Perguntas Freqüentemente Colocadas na Linux-BR (FAQ)

Original de Maio - 1996, Ian Jackson <ijackson@gnu.ai.mit.edu>
Versão em português mantida por Alfredo K. Kojima <kojima@inf.ufrgs.br>

Esta é a lista de perguntas freqüentemente colocadas (FAQ) sobre Linux, o Unix gratuito para 386/486/586 [Veja P1.1 para mais informações]. Leia-a juntamente com os documentos HOWTO, disponíveis em:

ftp.funet.fi (128.214.6.100) : /pub/OS/Linux/doc/HOWTO
tsx-11.mit.edu (18.172.1.2) : /pub/linux/docs/HOWTO
sunsite.unc.edu (152.2.22.81) : /pub/Linux/docs/HOWTO

e em outros espelhos -- veja P2.5 "Onde pego material sobre Linux por FTP?".
Veja P2.1 "Onde eu pego os HOWTOs e outra documentação?" para uma lista dos HOWTOs e outras informações. O INFO-SHEET e META-FAQ, disponíveis no mesmo lugar, também indicam outras fontes de informações sobre Linux.

Este documento baseia-se no LINUX FREQUENTLY ASKED QUESTIONS WITH ANSWERS, mantida por Ian Jackson<ijackson@gnu.ai.mit.edu>. Além da tradução completa
do documento original em inglês, ela contém perguntas freqüentes na lista de discussões Linux-BR [Veja Quais são as listas de discussões sobre Linux?
para obter informações sobre como assinar a lista]

Os documentos do Linux Documentation Project estão em sunsite.unc.edu em /pub/Linux/docs/LDP. Esses documentos (existem mais em preparação) são indispensáveis para os novatos ou para serem usados como referência.

Por favor, dê uma olhada nesses documentos e neste FAQ, especialmente em P12.1 "Vocês ainda não responderam à minha pergunta!", antes de enviar sua pergunta a um grupo de discussão ou à Linux-BR.

Veja P13.2 Formatos em que este FAQ está disponível para detalhes de onde pegar as versoes em PostScript, HTML (WWW) e ASCII puro deste documento, bem como a versão original em inglês em diversos formatos.

Uma nova versão deste documento será criada mensalmente. Se esta cópia tiver mais de um mês, ela pode estar desatualizada.

Índice

P4.9. Há algum produto como Stacker ou Doublespace para Linux?
P4.10. Minhas partições ext2fs são verificadas toda vez que reinicio o sistema.
P4.11. Meu sistema de arquivos principal está "read-only"!
P4.12. Tem um arquivo /proc/kcore enorme! Posso apagá-lo?
P4.13. Minha controladora AHA1542C não funciona com Linux.

Seção 5. Portando, compilando e obtendo programas

P5.1. Como porto XXX para Linux?
P5.2. O que é ld.so e onde posso obtê-lo?
P5.3. Alguém já portou / compilou / escreveu XXX para Linux?
P5.4. É possível usar um programa ou compilador compilado para um 486 em um 386?
P5.5. O que gcc -O6 faz?
P5.6. Onde estão <linux/*.h> e <asm/*.h>?
P5.7. Eu recebo erros quando tento compilar o kernel
P5.8. Como faço uma bilioteca dinâmica?
P5.9. Meus executáveis são (muito) grandes.
P5.10. O Linux suporta "threads" ou processos "lightweight"?
P5.11. Onde posso obter `lint' para Linux?

Seção 6. Soluções para problemas comuns diversos.

P6.1. free não funciona.
P6.2. Meu relógio está totalmente errado.
P6.3. Scripts setuid parecem não funcionar.
P6.4. A memória livre mostrada pelo free fica diminuindo.
P6.5. Linux fica muito lento quando coloco mais memória.
P6.6. Alguns programas (xdm) não me deixam "logar".
P6.7. Alguns programas me deixar "logar" sem senha.
P6.8. Minha máquina roda muito lentamente quando eu executo GCC / X / ...
P6.9. Eu só consigo me "logar" como root.
P6.10. Minha tela está cheia de caracteres esquisitos ao invés de letras.
P6.11. Houve um desastre em meu sistema e não consigo me logar para consertá-lo.
P6.12. Descobri uma falha de segurança enorme no rm!
P6.13. lpr e/ou lpd não estão funcionando.
P6.14. A hora dos arquivos de partições msdos estão gravados incorretamente.
P6.15. Como faço o LILO carregar o arquivo vmlinux?

Seção 7. Como faço isso ou aquilo...?

P7.1. Como faço para rolar a tela para cima em modo texto?
P7.2. Como troco de console virtual? Como os habilito?
P7.3. Como configuro o timezone (fuso-horário)?
P7.4. Que versão de Linux e que máquina estou usando?
P7.5. Como posso habilitar ou desabilitar core dumps?
P7.6. Como atualizo/recompilo meu kernel?
P7.7. Posso ter mais de 3 portas seriais compartilhando interrupções?

Questão 1.1. O que é Linux?

Linux é um clone do Unix escrito por Linus Torvalds com a assistência de um grupo de hackers espalhados pela Internet. Pretende-se que ele conforme com o padrão POSIX.

Ele faz tudo o que você esperaria de um Unix moderno e completo, incluindo multi-tarefa real, memória virtual, shared libraries (bibliotecas de "linkagem" dinâmica), carregamento por demanda, redes TCP/IP, alem de nomes de arquivos com até 255 caracteres, proteção entre processos (crash protection), suporte a UNICODE etc.

Ele roda principalmente em PCs baseados em 386/486/586, usando recursos de hardware da família de processadores 386 (segmentos TSS etc.) para implementar essas funções. Ports (versões modificadas para outro sistema) para outras arquiteturas estão a caminho. [P1.4. Para quais plataformas ele foi portado?].

Veja o INFO-SHEET do Linux [P2.1. Onde pego os HOWTOs e outra documentação?
] para mais detalhes.

O kernel (núcleo do sistema operacional) é distribuído sob os termos do GNU General Public License - veja P1.6. O Linux está em Domínio Público? Tem Copyright? para mais detalhes.

Questão 1.2. Que programas ele suporta?

O Linux tem GCC, Emacs, X-Windows, todos os utilitários Unix padrão, TCP/IP (inclusive SLIP e PPP) e centenas de outros programas compiladas ou portadas para ele por diversas pessoas.

Existe um emulador de DOS (veja em tsx-11.mit.edu em /pub/linux/ALPHA/dosemu) que pode rodar o DOS e algumas (não todos)

aplicações DOS. Há rumores de que agora ele pode rodar o Windows 3.1 em modo avançado.

Um emulador iBCS2 (Intel Binary Compatibility Standard) para binários ELF de SVR4 e COFF de SVR3.2 está em um estágio consideravelmente avançado de desenvolvimento. Veja o arquivo /pub/linux/BETA/ibcs2/README em tsx-11.mit.edu.

trabalho no emulador para binários do MS Windows [P3.6. Eu posso rodar programas do Microsoft Windows no Linux?] esta progredindo; alternativamente, a equipe do emulador de DOS tem tido sucesso em rodar o próprio Windows dentro do dosemu - haverá um anúncio se e quando ele funcionar.

Para mais informações leia o INFO-SHEET, que é um dos HOWTOs [P2.1. Onde pego os HOWTOs e outra documentação?]. Veja também P5.1 Como eu porto XXX para o Linux?.

Algumas empresas têm software comercial disponível, incluindo Motif. Eles os anunciam no grupo comp.os.linux.announce --- tente dar uma olhada nos repositórios de mensagens [P2.9. Os newsgroups são arquivados em algum lugar?].

Questão 1.3. Ele roda no meu micro? Que hardware ele suporta?

Você precisa de um 386, 486 ou 586, com pelo menos 2Mb de RAM e um drive para experimentá-lo. Para fazer qualquer coisa útil, mais RAM (4Mb para instalar a maioria das distribuições, e 8Mb é altamente recomendado para rodar X) e um disco rígido são necessários.

VESA local bus e PCI são ambos suportados.

Existem problemas com máquinas que usam MCA (barramento proprietário da IBM), principalmente com o controlador de discos. Há um pré-lançamento para desenvolvedores para drives ESDI de PS/2 em invaders.dcrl.nd.edu em /pub/misc/linux. Alguns controladores SCSI também funcionam, pelo que entendo. Existe trabalho em andamento para criar uma versão adaptada da distribuição Slackware. Para maiores detalhes tente perguntar para Arindam Banerji <axb@defender.dcrl.nd.edu> .

Linux roda em Laptops baseados na família 386, com X na maioria. Uma página relevante é http://www.cs.utexas.edu/users/kharker/linux-laptop/.

Para detalhes de exatamente quais PCs, placas de vídeo, controladores de disco etc. funcionam, leia o INFO-SHEET e o Hardware HOWTO [P2.1. Onde pego os HOWTOs e outra documentação?].

Existe um port do Linux para 8086, conhecido como Embeddable Linux Kernel Subset (ELKS). Este é um subconjunto de 16-bits do kernel do Linux, que irá ser usado principalmente em embeded systems{?}. Veja http://www.linux.org.uk/Linux8086 para mais informaçoes. Linux jamais rodará totalmente em um 8086 ou 286, pois ele requer recursos de gerenciamento de memória e de chaveamento de tarefas não encontrado nesses processadores.

Versões recentes do Linux (1.3.35 e posteriores) suportam máquinas multiprocessadas (SMP), apesar dele ainda nao estar satisfatoriamente estável.

Questão 1.4. Para quais plataformas ele foi portado?

Um projeto para portar o Linux para sistemas baseados na família 68000, como Ataris e Amigas, está em andamento. Ele já tem qualidade beta e tem um servidor X. Há uma lista de discussão sobre linux-680x0 [P2.8. Quais são as listas de discussão sobre Linux?] e um FAQ Linux/68K em http://pfah.informatik.uni-kl.de:8000/pers/jmayer/linux68k-faq e em ftp://tsx-11.mit.edu/pub/linux/680x0/FAQ; há mais informações em http://www-users.informatik.rwth-aachen.de/~hn/linux68k.html. Há uma lista de discussões sobre o port do Atari - mande um mail para majordomo@phil.uni-sb.de com corpo da mensagem contendo somente subscribe atarix - e uma área de FTP em ftp.phil.uni-sb.de em /pub/atari/linux.

Existia um projeto para portar o Linux para Macintoshes baseados em 68000, mas sua área de FTP desapareceu recentemente e o projeto parece estar abandonado.

Existe um port para o PowerPC. Até 7 de julho (95), ele estava bastante precário e não podia nem se recompilar, além de só suportar placas Motorola 1603. Trabalho no Motorola Ultra, PowerStack, RS/6000 e máquinas NuBus continua. Se você desejar contribuir ao projeto assine a lista linux-ppc [P2.8 Que listas de discussão existem?]. Existe um FAQ em liber.stanford.edu em /pub/linuxppc/linux-ppc-FAQ e em http://liber.stanford.edu/linuxppc/linux-ppc-FAQ.html.

A Apple e OSF estão trabalhando em um port do Linux para o PowerMac, baseado no microkernel Mach da OSF. http://mklinux.apple.com/.

Existe um port para o DEC Alpha/AXP de 64bits. Veja http://www.azstarnet.com/~ axplinux/. Também há uma lista de discussões em vger.rutgers.edu.

Ralf Baechle está trabalhando em um port para MIPS, inicialmente para o R4600 em máquinas Deskstation Tyne. O Linux/MIPS FAQ está disponível em WWW
e na área do port para MIPS em ftp.waldorf-gmbh.de em /pub/linux/mips Também há um canal no servidor de mail do Linux Activists e uma lista linux-mips [P2.8. Quais são as listas de discussão sobre Linux?]. Pessoas interessadas podem mandar suas perguntas e ofertas de ajuda para linux@waldorf-gmbh.de.

Atualmente existem dois ports do Linux para a família ARM de processadores em andamento; um deles é o ARM3, usado no Acorn A5000, e inclui drivers de I/O para o 82710/11, e o outro é para o ARM610, usado no Acorn Risc PC. O port para o Risc PC está atualmente em estágio médio, necessitando da reescrita do código de gerenciamento de memória. O port para o A5000 está em besta-teste restrito e poderá ser lançado em breve. Para informações mais recentes cheque comp.sys.acorn.misc, olhe veja no WWW em http://whirligig.ecs.soton.ac.uk/~rmk92/armlinux.html; se você quiser voluntarear contate Martin Ebourn mje@soton.ac.uk

David Miller está trabalhando em um port para o Sparc. Ainda está num

estágio bastante primário; pessoas dispostas a dedicar bastante tempo e com acesso a Sparcs para teste devem entrar em contato com davem@caip.rutgers.edu.

Nenhum dos ports acima poderá rodar binários do Linux/386.

O port Linux para a placa Aleph One 486 foi terminado e parece estável. Para maiores detalhes: http://www.ph.kcl.ac.uk/~amb /linux.html

Questão 1.5. De quanto espaço em disco ele precisa?

10Mb para uma instalação mínima, o suficiente para experimenta-lo e pouca coisa mais.

Você pode espremer uma instalação mais completa com X Windows em 80Mb. Instalar quase todo o Debian 0.93R6 ocupa uns 500Mb, incluindo algum espaço para arquivos de usuários e áreas para spool.

Questão 1.6. O Linux está em Domínio Público? Tem Copyright?

O copyright do kernel do Linux pertence a Linus Torvalds. Ele o colocou sob o GNU General Public License, o que basicamente significa que voce pode copiá-lo livremente, modifica-lo e distribuí-lo, mas você não pode impor quaisquer restrições na sua distribuição e ainda deve deixar o código fonte disponível. Isto não é o mesmo que domínio público (leia o FAQ sobre copyright que está disponível em rtfm.mit.edu em /pub/usenet/news.answers/law/Copyright-FAQ, para detalhes).

Detalhes completos estão no arquivo COPYING nos fontes do kernel do Linux (provavelmente em /usr/src/linux em seu sistema).

As licensas dos utilitários e programas que vêm com as instalações são variadas; muito desse código é do projeto GNU da Free Software Foundation, e também está sobre o GPL.

Note que discussões sobre o mérito ou não do GPL devem ser postadas a gnu.misc.discuss e não para os grupos comp.os.linux.

Questão 2.1. Onde pego os HOWTOs e outra documentação?

Dê uma olhada nesses lugares, e em outros que os espelham:

ftp.funet.fi (128.214.6.100) : /pub/OS/Linux/doc/HOWTO
tsx-11.mit.edu (18.172.1.2) : /pub/linux/docs/HOWTO
sunsite.unc.edu (152.2.22.81) : /pub/Linux/docs/HOWTO
No Brasil:
linux.if.usp.br (143.107.251.4) :
/pub/mirror/sunsite.unc.edu/pub/Linux/docs/HOWTO
farofa.ime.usp.br (143.107.251.4) : /pub/linux/docs/HOWTO

Para uma lista completa de sítios de FTP veja P2.5. Onde eu pego material sobre Linux por FTP?

Se você não tem acesso a FTP tente usar servidores de FTP-por-mail em ftpmail@decwrl.dec.com, ftpmail@doc.ic.ac.uk ou ftp-mailer@informatik.tu-muenchen.de.

Uma lista completa de HOWTOs está disponível no arquivo HOWTO.INDEX no diretório docs/HOWTO dos sítions de FTP, ou na Web em http://sunsite.unc.edu/mdw/HOWTO/HOWTO-INDEX.html

Esta é uma lista (provavelmente incompleta) dos HOWTOs:

Linux INFO-SHEET
Linux META-FAQ
Bootdisk HOWTO
Busmouse HOWTO
CDROM HOWTO
Commercial HOWTO
Cyrillic HOWTO
DOSEMU HOWTO
Danish HOWTO
Distribution HOWTO
ELF HOWTO
Ethernet HOWTO
Firewall HOWTO
Ftape HOWTO
German HOWTO
HAM HOWTO
Hardware HOWTO
Installation HOWTO
JE HOWTO
Kernel HOWTO
MGR HOWTO
Electronic Mail HOWTO
NET-2 HOWTO
NIS HOWTO
News HOWTO
PCI-HOWTO
PCMCIA HOWTO
Portuguese HOWTO
PPP HOWTO
Printing HOWTO
SCSI HOWTO
SCSI Programming HOWTO
Serial HOWTO
Sound HOWTO
Term HOWTO
Tips HOWTO
UPS HOWTO
UUCP HOWTO
XFree86 HOWTO

Outros documentos desses estão sempre em preparação. Se não encontrar a informação que você precisa em um destes documentos procure em outros diretórios próximos nos sítios de FTP. Além destes HOWTOs ainda existem os

mini HOWTOs, no diretório docs/HOWTO/mini.

O arquivo WRITING contém informações sobre como escrever um novo HOWTO.

Os HOWTOs são coordenados por Greg Hankins gregh@cc.gatech.edu.

Os livros produzidos pelo Linux Documentation Project estão disponíveis em /pub/Linux/docs/LDP em sunsite.unc.edu. Por favor leia-os se você for novo em Unix e Linux. Sobretudo o Installation and Getting Started Guide. Os livros produzidos até o momento são:

The Linux Documentation Project manifesto
Installation and Getting Started Guide
The Kernel Hacker's Guide
Network Administration Guide
Linux System Administrator's Guide

Questão 2.2. O que tem na World Wide Web sobre Linux?

Matt Welsh mantém Home Page do Linux Documentation Project, em http://sunsite.unc.edu/mdw/linux.html.

Esta página se refere a todos os FAQs e HOWTOs, tanto em formato HTML quanto em outros.

A página da Linux-BR está em http://www.br.freebsd.org/linux-br/index.html. Ela é mantida pelo Eduardo Maçan e possui além de links para outras páginas, um repositório de todas as mensagens vinculadas à lista e versões atualizadas deste documento.

Questão 2.3. Que grupos de discussão (newsgroups) sobre Linux existem?

Existem 10 newsgroups Usenet internacionais devotadas ao Linux.

comp.os.linux.announce é um grupo de anúncios moderado; você deve lê-lo se pretende usar Linux. Submissões a este grupo devem ser mandadas para linux-announce@news.ornl.gov.

comp.os.linux.answers Contém todos os FAQs, HOWTOs e outros documentos importantes. Assine este grupo também.

Os outros grupos na hierarquia comp.os.linux.* também são recomendados -- algums problemas comuns não respondidos neste FAQ podem estar nos newsgroups. Esses grupos são comp.os.linux.setup, comp.os.linux.hardware, comp.os.linux.networking, comp.os.linux.x, comp.os.linux.development.apps, comp.os.linux.development.system, comp.os.linux.advocacy e comp.os.linux.misc.

Lembre-se que como o Linux é um clone do Unix, a maioria do material em comp.unix.* e comp.windows.x.* sera relevante. Excetuando-se de algumas considerações de hardware e algum assunto obscuro ou muito técnico e de baixo-nível, você achará que esses grupos são o lugar certo para começar.

Por favor leia Vocês ainda não responderam à minha pergunta! antes de
postar, e certifique-se de postar no grupo certo -- veja `Welcome to the
comp.os.linux.* hierarchy' que é postado a cada duas semanas em
comp.os.linux.announce, comp.os.linux.answers e outros grupos.

Crossposting (mandar a mesma mensagem para mais de um grupo de discussão) em
grupos comp.os.linux.* diferentes é raramente uma boa idéia.

Podem haver grupos de discussão locais à sua instituição ou área -
verifique-os antes.

Os grupos comp.os.linux.development, comp.os.linux.admin and
comp.os.linux.help foram substituídos em uma reorganização de grupos
recente. Não os use mais.

Veja também P2.7. Eu nao tenho acesso à Usenet. Onde consigo informações?
P2.8. Quais são as listas de discussão sobre Linux?

Questão 2.4. Como instalo o Linux?

Existem várias "releases" de Linux pré-empacotadas disponíveis, incluindo
Debian, Red Hat e Slackware. Cada um contém o software que você precisa para
rodar Linux, prontos para instalar e usar. Detalhes de quais softwares estão
incluídos e como instalá-los varia de versão a versão.

O Mini-linux e' um pacote pequeno (4 disquetes) -- mas que inclui TCP/IP,
X-Windows e outras coisas -- apropriado para pessoas que querem experimentar
o Linux sem reparticionar seus discos rigidos.

Voce deve ler o Installation HOWTO para mais detalhes em como instalar o
pacote Slackware. O Red Hat e Debian são mais recentes, têm menos bugs e vêm
com esquemas de instalação sofisticados, mas não são tão populares quanto o
Slackware e não contém uma gama de software tão larga.

Todos esses pacotes estão disponíveis por FTP anônimo em vários sítios de
FTP P2.5. Onde eu pego material sobre Linux por FTP? Também existe um garnde
número de outros pacotes distribuídos menos globalmente, mas atendem melhor
a necessidades locais e nacionais (como suporte melhor à
internacionalização).

Questão 2.5. Onde eu pego material sobre Linux por FTP?

Os três sítios principais de Linux são:

ftp.funet.fi (Finland, 128.214.6.100) : /pub/OS/Linux
sunsite.unc.edu (US, 152.2.22.81) : /pub/Linux
tsx-11.mit.edu (US, 18.172.1.2) : /pub/linux

O melhor lugar para se pegar versões novas de kernels é ftp.cs.helsinki.fi
em /pub/Software/Linux/Kernel; Linus Torvalds disponibiliza as versões mais
recentes do kernel nesse lugar.

A distribuição Debian é disponível em ftp.debian.org e a distribuição Red Hat em ftp.redhat.com

O conteúdo destes sítios são espelhados (copiados, em geral diariamente) por outros sítios. Por favor use um perto de você -- será mais rápido para você e nais fácil para a rede.

src.doc.ic.ac.uk : /packages/Linux (UK)
sunacm.swan.ac.uk : /pub/Linux (UK)
ftp.ibp.fr : /pub/linux (França)
ftp.cc.gatech.edu : /pub/linux (EUA - sudeste: Suranet)
wuarchive.wustl.edu : /systems/linux (EUA)
uiarchive.cso.uiuc.edu : /pub/systems/linux (EUA)
ftp.cdrom.com : /pub/linux (EUA)
ftp.informatik.tu-muenchen.de : /pub/comp/os/linux (Alemanha)
ftp.ibr.cs.tu-bs.de : /pub/os/linux (Alemanha)
ftp.dfv.rwth-aachen.de : /pub/linux (Alemanha)
ftp.informatik.rwth-aachen.de : /pub/Linux (Alemanha)
bond.edu.au : /pub/OS/Linux (Austrália)
ftp.cc.monash.edu.au : /pub/linux (Austrália)
ftp.dstc.edu.au : /pub/Linux (Austrália: Queensland)
ftp.sun.ac.za : /pub/linux (África do Sul)
ftp.inf.utfsm.cl : /pub/Linux (Chile)
ftp.zel.fer.hr : /pub/Linux (Croácia)
linux.if.usp.br : /pub/mirror/sunsite.unc.edu/Linux além de outros em
/pub/mirror com a distribuicao Debian.
lcmi.ufsc.br : /pub/diversos/linux (Brasil : Santa Catarina) Slackware
cesar.unicamp.br : /pub3/linux (Brasil : São Paulo) Slackware
ftp.ime.usp.br : /pub/linux (Brasil : São Paulo) Slackware
ftp.ufpr.br : /pub/Linux/ (Brasil : Paraná) Slackware

Nem todos eles espelham todo conteúdo dos sítios "originais", e alguns têm material que os sítios "originais" não têm.

Questão 2.6 Eu não tenho acesso a FTP. Como eu consigo o Linux?

O jeito mais fácil é talvez achar um amigo com acesso a FTP. Se existir um Grupo de Usuários de Linux perto de você, eles poderão ajudá-lo.

Se você tiver uma conexão de email razoavelmente boa, você poderia tentar os servidores de FTP-por-mail em ftpmail@decwrl.dec.com, ftpmail@doc.ic .ac.uk ou ftp-mailer@informatik.tu-muenchen.de.

Linux também é disponível por correio tradicional em disquetes, CD-ROM e fita. Installation HOWTO e o arquivo /pub/Linux/docs/distributions em sunsite.unc.edu, contém informaçoes sobre essas distribuições.

Várias empresas que comercializam CDs do Linux têm páginas na WWW. Eles têm um preço razoável e despacham para outros países. Veja em http://www.ix.de/ix/linux/bookmarks/companies.html para alguns links de algumas dessas empresas.

Você ainda pode tentar a lista de BBSs de Zane Healy healyzh@holonet.net que

é regularmente postado (dia 1o e 15 de cada mês) em comp.os.linux.announce e ocasionalmente na Fidonet e RIME UNIX echoes.

Questão 2.7. Eu não tenho acesso à Usenet. Onde consigo informações?

Um resumo do comp.of.linux.announce pode ser obtido mandando e-mail contendo subscribe para linux-announce-REQUEST@news-digests.mit.edu. Recomenda-se que você assine esta lista, pois ela contém informações importantes e documentação sobre Linux.

Lembre-se de adicionar -request depois do nome do newsgroup (linux-anounce-request) para requisitar a sua assinatura. Se -request for omitido, a sua mensagem será postada no grupo de discussões.

Questão 2.8 Quais são as listas de discussão sobre Linux?

Os desenvolvedores do Linux usam principalmente o servidor Majordomo em majordomo@vger.rutgers.edu . Mande uma mensagem com lists no corpo da mensagem para receber uma lista das listas lá; adicione uma linha com help para receber o arquivo de ajuda do Majordomo, com instruções para inscrição e desinscrição.

Note que a maioria dessas listas são usadas pelos desenvolvedores do Linux para falarem sobre assuntos técnicos e desenvolvimento futuro. Elas não devem ser usadas para perguntas de usuário novatos.

Existe a lista linux-newbie, onde `nenhuma pergunta é idiota demais'; infelizmente parece que poucos usuários experientes a lêem. O tráfego é bastante baixo.

No Brasil você pode assinar a Linux-BR mandando um e-mail contendo subscribe linux-br <seu nome> para listproc@iqm.unicamp.br.

Questão 2.9 Os newsgroups são arquivados em algum lugar? (inclui Linux-BR)

sunsite.unc.edu contém mensagens do comp.os.linux.announce, em sunsite.unc.edu em /pub/Linux/docs/linux-announce.archive . Eles são espelhados de /usenet em src.doc.ic.ac.uk.

Existe um repositório de fácil acesso do comp.os.linux.announce na World Wide Web em http://www.leo.org/archiv/linux/archiv/ann_index.html.

O repositório das mensagens da Linux-BR está em http://www.br.freebsd.org/linux-br/LINUX-BRmails.html

Questão 3.1. Eu posso instalar o Linux junto com o DOS? OS/2? 386BSD? Win95?

Sim. O Linux usa o esquema padrão de particionamento do PC, assim ele pode compartilhar o seu disco com outros sistemas operacionais. No entanto, muitos desses outros sistemas operacionais são mal feitos: o FDISK e FORMAT do DOS podem destruir dados em uma partição Linux, porque às vezes eles usam incorretamente os dados do setor de boot da partição, ao invés da tabela de partição do disco. Para previnir isso, zere o começo da partição nova no

Linux, antes de usar o MSDOS ou outra coisa para formatá-la. Digite:

```
dd if=/dev/zero of=/dev/hdXY bs=512 count=1
```

onde hdXY é a partição relevante (por ex.: hda1 para a primeira partição do primeiro disco IDE).

O Linux pode ler e escrever arquivos em partições FAT do DOS e OS/2 e em disquetes usando o suporte interno do kernel ou o mtools. Existe uma versão alpha de suporte do kernel para o sistema VFAT usado pelo Widows 95 e Windows NT; ele está disponível em /pub/multimedia/linux/vfat/vfat-0.3.0.tgz. Ele foi integrado ao kernel padrão na versão 1.3.60.

Veja P1.2. Que programas ele suporta? para detalhes e status dos emuladores de DOS, MS Windows e programas para System V.

Veja também P3.5. Eu posso acessar sistemas de arquivos como o FFS do BSD, UFS do SysV, Mac, Amiga etc?

ftp://sunsite.unc.edu/pub/Linux/docs/howto/mini/Win95+Win+Linux, ftp://sunsite.unc.edu/pub/Linux/docs/howto/mini/Linux+DOS+Win95+OS2 e ftp://sunsite.unc.edu/pub/Linux/docs/howto/mini/Linux+OS2+DOS contem instrucoes especificas sobre instalacao de varios sistemas operacionais em uma mesma maquina.

Questão 3.2. Como eu acesso arquivos na minha partição DOS ou em disquetes?

Use o filesystem DOS, i.e. digite, por exemplo:

```
mkdir /dos
mount -t msdos -o conv=text,umask=022,uid=100,gid=100 /dev/hda3 /dos
```

Se for um disquete, não esqueça de umountar ele antes de tirá-lo do drive, ou dados serão perdidos.

Você pode controlar a aparência dos arquivos no filesystem no Linux quanto à conversão automática de fim de linha CR/LF para CR, permissões e propriedade usando conv=text/binary/auto, umask=nnn, uid=nnn e gid=nnn. Se você montar seu filesystem DOS colocando-o no /etc/fstab você pode configurar as opções trocando o defaults pelas suas opções separadas por vírgulas.

Você também pode usar o `mtools', disponível tanto em forma binária e em código fonte nos sítios FTP -- veja P2.5. Onde eu pego material sobre Linux por FTP? Algumas distribuições já vêm com ela.

Existe um "patch" para kernel (conhecido como fd-patches), que permitem o uso de disquetes com números de trilhas e/ou setores diferentes do padrão; esse patch é incluído na série 1.1 de kernels em teste alpha.

Questão 3.3. Eu posso usar drives do DOS com Stacker/Double Space/etc?

Não de maneira facil. Você pode acessá-los pelo emulador de DOS [P1.2 Que

programas ele suporta?], mas para acessá-los no Linux ou pelo mtools é mais difícil. Existe um módulo para o kernel do Linux que pode ler volumes comprimidos, mas não pode escrever neles. Procure o pacote ths na sunsite.unc.edu em /pub/Linux/system/Filesystems

Questão 3.4. Eu posso acessar partições HPFS do OS/2 no Linux?

Sim, mas no momento apenas para leitura. Para usá-lo o kernel deve ser compilado com o suporte para ele habilitado [P7.6. Como atualizo/recompilo meu kernel?]. Aí é só montá-lo com o comando mount, por ex.:

```
mkdir /hpfs
mount -t hpfs /dev/hda5 /hpfs
```

Questão 3.5. Eu posso acessar sistemas de arquivos como o FFS do BSD, UFS do SysV, Mac, Amiga etc?

Há um filesystem Amiga precário e em alpha-teste em /pub/Linux/patches/amigaffs.tar.Z. Mais informações no arquivo affs-readme do arquivo tar.

Kernels recentes contém suporte para o UFS no System V, Coherent e Xenix.

Existe suporte do kernel em alpha, por enquanto somente para leitura, para o UFS do 4.2BSD, em sunsite.unc.edu em /pub/Linux/ALPHA/ufs.

Ainda não existe suporte para o resto.

Questão 3.6. Eu posso rodar programas do Microsoft Windows no Linux?

Ainda não. Existe um projeto, o WINE, para construir um emulador de MS Windows para o Linux, mas ainda não está disponível para usuários. Não pergunte sobre ele, a não ser que você ache que pode contribuir; procure os relatórios de status em comp.emulators.ms-windows.wine.

No momento, se você precisar usar algum programa do MS Windows, o melhor é provavelmente rodar o Windows direto sob o DOS. LILO (o bootloader do Linux) possui um menu para escolher o sistema a entrar --- veja a documentação dele para mais detalhes.

Questão 3.7. Como eu posso inicializar o Linux pelo Boot Manager do OS/2?

1. Crie uma partição usando o FDISK do OS/2. (Não serve o fdisk do Linux).
2. Formate a partição no OS/2, em FAT ou HPFS. Isto é para fazer o OS/2 saber que a partição está formatada. (Este passo não é necessário no OS/2 "warp" 3.0).
3. Adicione a partição no Boot Manager.
4. Inicie o Linux, e crie um sistema de arquivos na partição usando mkfs -t ext2 ou mke2fs. A essa altura você pode usar o fdisk do Linux para trocar o código de tipo da partição para 83 (Linux Native) -- isso pode ajudar alguns scripts de instalação automática a achar a partição certa

para usar.
5. Instale o Linux na partição.
6. Instale o LILO na partição -- não o faça no master boot record do disco rígido. Isso instala o LILO como um bootloader secundário na partição do Linux, para carregar o kernel especificado no arquivo de configuração do LILO. Para isso coloque:

   boot = /dev/hda2

   (onde /dev/hda2 é a partição de onde você quer "bootar") no seu arquivo /etc/lilo/config ou /etc/lilo.conf
7. Certifique-se de que a partição do Boot Manager esteja marcada como ativa, para poder usar o Boot Manager para escolher o que carregar.

Questão 3.8. Como eu faço para usar a mesma partição de swap no Windows e no Linux?

Veja o Mini-HOWTO de H. Peter Anvin hpa@yggdrasil.com sobre esse assunto na sunsite.unc.edu em /pub/Linux/docs/HOWTO/mini/Swap-Space.

Questão 4.1. Como posso fazer o Linux funcionar com meu disco "grande" de mais de 1024 cilindros lógicos?

Se seu disco for uma unidade IDE ou EIDE você deve ler o arquivo /usr/src/linux/drivers/block/README.ide (parte do código-fonte do "kernel" do Linux). Esse README contém várias sugestões úteis sobre unidades de disco IDE.

Discos SCSI são acessados através de números lineares de blocos; o BIOS cria alguma simulação de cilindros/cabeças/setores "lógicos" para atender ao DOS.

DOS, em geral, não estará em condições de acessar partições que se estendam além dos 1024 cilindros lógicos, e irá tornar a iniciação de um "kernel" Linux a partir dessas partições no mínimo problemática.

Você ainda poderá usar essas partições para Linux ou quaisquer outros sistemas operacionais que acessem a controladora de disco diretamente.

Recomendo criar no mínimo uma partição Linux contida inteiramente dentro do limite de 1024 cilindros lógicos e iniciar o sistema dessa partição; as outras partições estarão "OK".

Questão 4.2. Como posso reaver (undelete) arquivos apagados?

Em geral, isso é muito dificil de conseguir em sistemas Unix em razão da sua natureza multitarefa. Funções de recuperação estão em desenvolvimento, mas não espere grandes resultados.

Há uns poucos pacotes disponíveis que funcionam oferecendo novos comandos

para apagamento (e em alguns casos cópia) que movem os arquivos "apagados" para um diretório "lixeira"; esses arquivos podem ser entao recuperados, até que sejam apagados automaticamente por processos em segundo plano..

Outra alternativa é a busca direta na própria unidade de disco que contém o sistema de arquivos em questão. Esse é um trabalho difícil, e você precisa estar como root para isso.

Questão 4.3. Há algum defragmentador para ext2fs e outros sistemas de arquivos?

Sim. Há um defragmentador de sistemas de arquivos Linux para ext2, minix e antigos sistemas de arquivos ext disponível em sunsite.unc.edu como system/Filesystems/defrag-0.6.tar.gz.

Usuários do sistema ext2 provavelmente não precisarão de defragmentação pois ext2 contém código extra para manter a fragmentação reduzida, mesmo em sistemas de arquivos muito cheios.

Questão 4.4. Como posso formatar um disquete e criar nele um sistema de arquivos?

Para um disquete de 3,5 polegadas:

```
fdformat /dev/fd0H1140
mkfs -t ext2 -m 0 /dev/fd0H1140 1440
```

Para um disquete de 5,25 polegadas use fd0H1200 e 1200, respectivamente. Para a unidade de disco 'B' use fd1 em lugar de fd0. Detalhes completos do que se pode fazer com unidades de disco flexível podem ser encontrados na Lista de Dispositivos Linux (Linux Device List) [P2.1. Onde pego os HOWTOs e outra documentação?]. Você pode precisar 'rodar' mk2efs diretamente em lugar de mkfs -t ext2. A opção -m 0 diz a mkfs.ext2 para não reservar espaço no disco para o superusuário -- habitualmente os 10% finais são reservados para ele. O primeiro comando formata o disquete em baixo nivel (ou fisicamente); o segundo cria um sistema de arquivos vazio nele. Depois de fazer isso você pode montar o disquete como um disco rígido e simplesmente usar cp e mv em arquivos etc.

Questão 4.5. Estou recebendo mensagens estranhas sobre "inodes", "blocks", e coisas desse gênero.

Você provavelmente tem um sistema de arquivos corrompido, talvez provocado por não se preparar o Linux da maneira adequada antes de desligar o computador ou reiniciá-lo. Você precisa de um programa 'shutdown' recente para fazer isso -- por exemplo, aquele incluído no pacote 'util-linux', disponível em sunsite e tsx-11.

Se você tiver sorte, o programa fsck (ou e2fsck ou xfsck como for adequado, se voce não possui o 'front-end' automático para fsck) terá condições de reparar seu sistema de arquivos; caso contrário, o sistema de arquivos estará destruído, e você terá de reiniciá-lo com mkfs (ou mke2fs, mkxfs etc.) e restaurá-lo do 'backup'.

Note bem: não tente verificar um sistema que está montado como sendo de leitura e escrita -- isso inclui a partição da raiz -- se você não vê

```
VFS: mounted root ... read-only
```

no momento da iniciação do sistema.

Questão 4.6. Minha área de paginação (swap) não funciona.

Quando você dá partida ao sistema (ou ativa a paginação manualmente) você deve ver

```
Adding Swap: NNNNk swap-space
```

Se você não vê qualquer mensagem, provavelmente está faltando swapon -av (o comando para ativar a paginação) em seu arquivo /etc/rc.local ou /etc/rc.d/* (os 'scripts' de iniciação do sistema) ou você esqueceu de colocar a entrada correta em /etc/fstab:

```
/dev/hda2      none          swap          sw
```

por exemplo.

Se você vê

```
Unable to find swap-space signature
```

você esqueceu de usar mkswap. Veja a página do man(ual) para detalhes; mkswap funciona de maneira análoga ao mkfs.

Verifique o documento Ins tallation HOWTO para instruções detalhadas para se configurar uma área de paginação (swap).

Questão 4.7. Como posso desinstalar LILO e voltar a reiniciar meu sistema pelo DOS?

Se você está utilizando DOS (MS-DOS 5.0 ou posterior, ou OS/2), digite FDISK /MBR (o que não está documentado). Isso vai restaurar um 'Master Boot Record' padrão do MS-DOS. Se você tem DR-DOS 6.0, acione FDISK da maneira habitual e selecione a opção 'Re-Write Master Boot Record'.

Se você não tem DOS 5 ou DR-DOS, vai precisar do setor de 'boot' original que LILO salvou quando você o instalou pela primeira vez. Você guardou aquele arquivo, não é? E provavelmente chamado boot.0301 ou algo assim. Digite

```
dd if=boot.0301 of=/dev/hda bs=512 count=1
```

(ou sda se você está utilizando um disco SCSI). Isso pode também destruir sua tabela de partições, portanto, cuidado! Se você está desesperado, pode usar

```
dd if=/dev/zero of=/dev/hda bs=512 count=1
```

o que irá apagar seu setor de 'boot' e sua tabela de partições completamente; você pode então reformatar o disco usando o formatador de sua preferência; entretanto, isso tornará o conteúdo dos seus discos inacessível -- você vai perder tudo, a menos que seja um 'expert'.

Note que o MBR do DOS inicia qualquer (única!) partição que esteja marcada como ativa: você pode precisar usar fdisk para ligar e desligar as marcas de ativa das partições convenientemente.

Questão 4.8. Porque não posso usar fdformat, exceto como root?

A chamada de sistema necessária para formatar um disco flexível somente pode ser acionada pelo usuário root, quaisquer que sejam as permissões de /dev/fd0*. Se você quiser que qualquer usuário pssa formatar um disco flexível, tente conseguir o programa fdformat2; ele contorna os problemas por ser 'setuid' para root.

Questão 4.9 Há algum produto como Stacker ou Doublespace para Linux?

No presente, nenhum dos sistemas de arquivos Linux pode fazer compressão.

Há um programa chamado Zlibc que permite a aplicações existentes ler transparentemente arquivos comprimidos ('GNU-zipped') como se eles não estivessem comprimidos. Após sua instalação, você pode comprimir arquivos com gzip e os programas ainda poderão encontrá-los, sem necessidade de mudança nos mesmos. Procure em sunsite.unc.edu, diretóorio < code>/pub/Linux/libs. O autor é Alain.Knaff@imag.fr.

Há um controlador de dispositivos blocados (block device driver) com recursos de compressão, que pode prover compressão 'on the fly', independente do sistema de arquivos, no kernel. É chamado 'DouBle'. Há uma distribuição (em forma apenas de código-fonte) em sunsite.unc.edu, diretório /pub/Linux/kernel/patches/diskdrives; o autor é Jean-Marc Verbavatz jmv@receptor.mgh.harvard.edu. Note que, devido ao fato de comprimir 'inodes' (informação administrativa) e diretórios, além de conteúdo de arquivos, qualquer problema de corrupção pode ser grave.

Há também um pacote disponível chamado tcx ('Transparent Compressed Executables') que permite manter executáveis pouco usados comprimidos e descomprimi-los temporariamente quando você precisar deles. Voce encontrará tcx nos sites FTP Linux [P2.5. Onde eu pego material sobre Linux por FTP?]; ele também foi anunciado em comp.os.linux.announce. Nota: este programa não é o mesmo que gzexe, que é uma implementação inferior do mesmo conceito.

Questão 4.10. Minhas partições ext2fs são verificadas toda vez que reinicio o sistema.

Veja P9.12 'EXT2-fs: warning: mounting unchecked filesystem'

Questão 4.11 Meu sistema de arquivos raiz está "read-only"!

Remonte-o. Se /etc/fstab estiver correto você pode simplesmente fazer

mount -n -o remount /

Se /etc/fstab está errado você deve fornecer o nome do dispositivo e possivelmente o tipo, e.g.:

mount -n -o remount -t ext2 /dev/hda2 /.

Para compreender porque seu sistema ficou nesse estado, veja P9.12. 'EXT2-fs: warning: mounting unchecked filesystem'.

Questão 4.12. Tem um arquivo /proc/kcore enorme! Posso apagá-lo?

Nenhum dos arquivos em /proc estão realmente lá -- eles são todos arquivos fictícios, criados pelo kernel, para dar a você informações sobre o sistema, e não consomem nenhum espaço em disco.

/proc/kcore é como um 'apelido' para a memória em seu computador; seu tamanho é igual à quantidade de RAM que ele possui, e se você pedir para lê-lo como um arquivo, o kernel executa leituras da memória.

Questão 4.13. Minha controladora AHA1542C não funciona com Linux.

A opção para permitir discos com mais de 1024 cilindros somente é requerida como uma alternativa para uma deficiência do DOS e deve ser *desligada* sob Linux. Para antigos kernels Linux você precisa desligar a maioria das opções do 'BIOS avançado' -- todas menos a opção de procurar pelo barramento (bus) por dispositivos iniciáveis.

Questão 5.1. Como porto XXX para Linux?

Em geral, programas Unix precisam de poucas modificações para serem portados. Simplesmente siga as instruções de instalação. Se você não souber e nem souber como encontrar as respostas para algumas das perguntas feitas pelo procedimento de instalação, você pode tentar adivinhar, mas isso tende a produzir programas defeituosos. Neste caso, é melhor você pedir para alguma outra pessoas fazer a portagem.

Se você tiver um programa para sistemas similares ao BSD, tente usar -I/usr/include/bsd e -lbsd nos lugares apropriados.

Questão 5.2. O que é ld.so e onde posso obtê-lo?

ld.so é um carregador de bibliotecas dinâmicas. Cada binário usando essas bibliotecas costumava ter cerca de 3K de código de iniciliazação que procurara e carregara essas bibliotecas. Agora, esse código foi colocado em uma biblioteca dinâmica especial, /lib/ld.so, que pode ser usado por todos os binários, gastando assim menos espaço em disco e facilitando atualizações.

Ele pode ser obtido de tsx-11.mit.edu em /pub/linux/packages/GCC e seus espelhos. A última versão, no momento da escrita deste, é

ld.so.1.7.11.tar.gz.

/lib/ld-linux.so.1 é a mesma coisa para ELF e vem no mesmo pacote do carregador a.out.

Questão 5.3. Alguém ja portou / compilou / escreveu XXX para Linux?

Primeiro, veja o Linux Software Map (LSM) --- ele está no diretório docs em sunsite.unc.edu e outros sites de FTP. Um mecanismo de procura é disponível por WWW em http://www.boutell.com/lsm/.

Verifique os sites FTP (veja P2.5 Onde eu pego material sobre Linux por FTP?') antes --- procure por palavras apropriadas nos arquivos find-ls ou INDEX. Cheque o Linux Projects Map (LPM), em ftp.ix.de em /pub/Linux/docs/Projects-Map.gz.

Se você não encontrar nada, você pode fazer download dos fontes e compilá-los você mesmo [P5.1. Como porto XXX para Linux?] ou se o pacote for muito grande e necessitar de algumas modificações mande uma mensagem para o grupo comp.os.linux.development.apps.

Se você compilar um programa grande, por favor faça upload dele para um ou mais sites FTP e envie uma mensagem para comp.os.linux.announce (submeta sua mensagem para linux-announce@news.ornl.gov).

Se você estiver procurando por um programa tipo aplicativo é provável que alguém já tenha escrito uma versão gratuita. Tente ler o FAQ em comp.sources.wanted para instruções de como achar fontes.

Você também pode ler o Projects-FAQ, disponível em /pub/Linux/docs/faqs/Projects-FAQ em sunsite.unc.edu.

Questão 5.4. É possível usar um programa ou compilador compilado para um 486 em um 386?

Sim, a menos que seja o kernel.

A opção -m486 do GCC, que é usado para compilar binários para máquinas 486, meramente faz algumas certas otimizações. Isso produz binários levemente maiores que rodam um pouco mais rápido em um 486. Porém eles ainda funcionam perfeitamente em um 386, apesar de uma pequena queda em sua performance.

Todavia, apartir da versão 1.3.35, o kernel usará instruções específicos de 486 em Pentium, se configurado para tais máquinas, impossibilitando a utilização deles em 386s.

O GCC pode ser configurado para 386 ou 486; a única diferença é que configurando-o para um 386, faz com que -m386 seja default e configurando-o para um 486, faz com que -m486 seja o default. Em qualquer caso, essa opção pode ser mudada em cada compilação ou editando /usr/lib/gcc-lib/i*-linux/n.n.n/specs.

Existe uma versão alpha do GCC que sabe fazer otimizações para 586, mas ele

ainda não é confiável, especialmente se forem pedidas muitas otimizações. O GCC para Pentium pode ser encontrado em tsx-11.mit.edu em /pub/linux/ALPHA/pentium-gcc. Recomendo usar o GCC normal para 486; diz-se que usando-se a opção -m386 produz código melhor para o Pentium, ou pelo menos um pouco menor.

Questão 5.5. O que gcc -O6 faz?

Atualmente o mesmo que -O2 (GCC 2.5) ou -O3 (GCC 2.6, GCC 2.7); qualquer número maior que aqueles, no momento faz a mesma coisa. Os Makefiles the kernels mais recentes usam -O2, você provavelmente deveria fazer o mesmo.

Questão 5.6. Onde estão <linux/*.h> e <asm/*.h>?

Estes estão nos diretórios /usr/include/linux e /usr/include/asm.

Porém eles devem ser links simbólicos para os fontes de seu kernel em /usr/src/linuc e não diretórios reais.

Se você não tiver os fontes do kernel faça download deles.

Então use rm para remover quaisquer arquivos desnecessários e ln para criar os links:

```
rm -rf /usr/include/linux /usr/include/asm
ln -sf /usr/src/linux/include/linux /usr/include/linux
ln -sf /usr/src/linux/include/asm /usr/include/asm
```

Hoje em dia /usr/src/linux/include/asm é um link simbólico para um deretório asm-<arch> específico para cada arquitetura - make symlinks criará os links simbólicos. Você também poderá precisar fazer make config em uma árvore de fontes do kernel recém "desempacotada" para criar <linux/autoconf.h>.

Questão 5.7. Eu recebo erros quando tento compilar o kernel

Certifique-se que /usr/include/linux e /usr/include/asm não são realmente diretórios e sim links simbólicos para /usr/src/linux/include/linux e /usr/src/linux/include/asm respectivamente.

Se necessário, apague-os usando rm e então use ln -s para criar os links como mostrado em P5.6. Onde estão <linux/*.h> e <asm/*.h>?'.

Lembre-se de que ao aplicar um patch a um kernel você deve usar a opção -p0 ou -p1; caso contrário o patch pode ser aplicado de forma incorreta. Veja a página de manual do patch para detalhes.

Se você estiver aplicando patches a um kernel mais recente que 1.1.44, você notará um novo diretório /usr/src/linux/include/asm-i386. O diretório asm lá deve ser removido. O "target" symlinks do Makefile fará eles serem links simbólicos para asm-i386 e arch/i386/boot respectivamente. A forma mais fácil de certificar-se de que não ocorram problemas deste tipo é fazer o download de todo kernel, ao invés de aplicar um ptach.

ld: unrecognised option `-qmagic' significa que você deve pegar um linker mais novo, de tsx-11.mit.edu em /pub/linux/packages/GCC, no arquivo binutils-2.6.0.2.bin.tar.gz.

Questão 5.8. Como faço uma bilioteca dinâmica?

Para ELF,

```
gcc -fPIC -c *.c
gcc -shared -Wl,-soname,libsoo.so.1 -o libfoo.so.1.0 *.o
```

Para a.out, pegue tools-n.nn.tar.gz de tsx-11.mit.edu, em /pub/linux/packages/GCC/src. Ele vem com documentação que lhe dirá o que fazer. Note que bibliotecas dinâmicas para a.out são bastante complicados.

Questão 5.9. Meus executáveis são (muito) grandes.

Com um compilador ELF (veja P8.2. O que é esse tal de ELF?) a causa mais comum de executáveis grandes é a falta de um link para a biblioteca .so apropriada que você está usando. Deve haver um link como libc.so para cada biblioteca como libc.so.5.2.18.

Com um compilador a.out (veja P8.2. O que é esse tal de ELF?) a causa mais comum do problema é o flag -g passado ao linkeditor (compilador). Ele produz um programa que é "linkado" estaticamente (além de informações para depuração no arquivo de saída), i.e. um que inclui uma cópia da bilioteca C, ao invés de usar uma copia ligada dinamicamente.

Outros fatores a serem investigados são -O e -O2 que habilitam otimização (veja documentação do GCC) e -s (ou o comando strip) que remove informações de símbolos do binário resultante (tornando a depuração totalmente impossível).

Você pode querer usar -N em executáveis muito pequenos (menores que 8K com a opcção -N), mas você não deve usá-lo a menos que esteja ciente das implicações sobre a performance, e definitivamente jamais em daemons.

Questão 5.10. O Linux suporta "threads" ou processos "lightweight"?

Assim como o modelo de multiprocessamento do Unix envolvendo processos "heavyweight", que é claro, são parte do kernel padrão do Linux, existem várias implementações de processos "lightweight" ou "threads", a maioria dos quais são pacotes genéricos para qualquer Unix.

Em sipb.mit.edu: /pub/pthread ou ftp.ibp.fr: /pub/unix/threads/pthreads. A documentação não está no pacote, mas está disponível em WWW em http://www.mit.edu:8001/people/proven/home_page.html. libcs mais novos do Linux contêm o fonte pthreads; o compilador GNU Ada em sunsite.unc.edu em /pub/Linux/devel/lang/ada/gnat-3.01-linux+elf.tar.gz contém binários feitos com esse código fonte.

Em ftp.cs.washington.edu: /pub/qt-001.tar.Z está QuickThreads. Mais informações podem ser encontrados no relatório técnico disponível no

mesmo local como /tr/1993/05/UW-CSE-93-05-06.PS.Z.
Em gummo.doc.ic.ac.uk: /rex está lwp, uma implementação bastante simples.
Em ftp.cs.fsu.edu: /pub/PART, está uma implementação em Ada. Esta é útil principalmente por conter vários papers em PostScript úteis para se aprender mais sobre "threads". Ele não é diretamente usável sob Linux.

Por favor contate os autores dos pacotes em questão para detalhes.

A versão 1,3,35 do kernel possui algum soporte para threads no kernel, mas esse código não foi bem testado.

Questão 5.11. Onde posso obter `lint' para Linux?

Funcionalidade aproximadamente equivalente está incluído no compilador C GNU (gcc) que é usado em sistemas Linux. Use a opção -Wall para habilitar a maioria dos avisos úteis extra. Veja o manual do GCC para mais detalhes (tecle contro-h seguido de i no Emacs e selecione a entrada para GCC).

Existe um programa gratuitamente disponível chamado `lclint' que faz quase a mesma coisa que o lint tradicional. O anúncio e o código fonte estão disponíveis em larch.lcs.mit.edu em /pub/Larch/lclint; no World Wide Web veja em http://larch-www.lcs.mit.edu:8001/larch/lclint.html.

Questão 6.1. free não funciona.

No Linux 1.3.57 e posteriores o formato do /proc/meminfo foi mudado para um formato que a implementação atual do free não entende.

Pegue a versão mais recente em sunsite.unc.edu em /pub/Linux/system/Status/ps/procps-0.99.tgz.

Questão 6.2. Meu relógio está totalmente errado

Existem dois relógios no seu computador. O relógio do hardware (CMOS) funciona mesmo quando o computador está deslidado e é usado quando o sistema inicia e pelo DOS (se você usá-lo). A hora do sistema, mostrada e mudada por date, é mantido pelo kernel enquanto o Linux está rodando.

Você pode ver a hora do relógio CMOS, ou acerta-lo pelo relógio do sistema ou vice-versa, com o programa /sbin/clock - veja man 8 clock.

Existem vários outros programas que podem corrigir os relógios para tempo de transferência ou ajuste sistemático pela rede. Alguns deles já podem estar instalados em seu sistema. Procure por adjtimex (faz ajustes), netdate e getdate (pegam a hora pela rede) ou xntp (daemon de rede completo e preciso).

Questão 6.3. Scripts setuid parecem não funcionar

Está correto. Esta característica foi deliberadamente desabilitada no kernel do Linux porque scripts setuid são quase sempre uma falha de segurança. Se

você quiser saber porquê, leia o FAQ do comp.unix.questions.

Questão 6.4. A memória livre mostrada pelo free fica diminuindo

O item `free' mostrado por free não inclui a memória usada como cache de buffer de disco - mostrado na coluna `buffers'. Se você quiser saber quanta memória realmente está livre, some a quantidade mostrada em `buffers' à de `free' - versões mais recentes de free mostram uma linha a mais com essa informação.

O cache de buffer de disco tende a crescer logo após o Linux iniciar, no momento em que você carrega programas usando mais arquivos, o conteúdo deles vai sendo guardado na cache. Após algum tempo o cache irá se estabilizar.

Questão 6.5. Linux fica muito lento quando coloco mais memória

Este é um sintoma comum de uma falha do cache com a memória adicional. O problema exato depende da sua placa-mãe.

Algumas vezes você precisa habilitar a cache de certas regiões na configuração do seu BIOS. Olhe no setup do CMOS e veja se existe alguma opção para usar a cache na nova área de memória e que esteja desabilitada. Aparentemente isso é bastante comum num 486.

Algumas vezes a RAM precisa estar em certos soquetes para que a cache possa atuar.

Algumas vezes você precisa ligar jumpers para habilitar a cache.

Algumas placas-mãe não usam cache para toda RAM se você tiver mais RAM do que os fabricantes esperavam. Geralmente uma cache de 256K resolverá o problema.

Em caso de dúvida consulte o manual de sua placa-mãe. Se você ainda não conseguir conserta-la porque a documentaçã não é adequada envie uma mensagem para comp.os.linux.hardware dando todos os detalhes - marca, modelo etc., assim, outros usuários de Linux poderão evitá-la.

Questão 6.6. Alguns programas (xdm) não me deixam logar

Provavelmente você está usando programas sem shadow password, mas está usando shadow passwords.

Se for o caso, você precisa pegar ou compilar uma versão para shadow password do(s) programa(s) em questão. Programas para shadow password podem ser encontrados em (dentre outros lugares):

tsx-11.mit.edu:/pub/linux/sources/usr.bin/shadow-*

Lá estão o códigos fonte; você provavelmente encontrará os binários em .../linux/binaries/usr.bin.

Questão 6.7. Alguns programas me deixar logar sem senha.

Você provavelmente tem o mesmo problema de P6.5. Linux fica muito lento quando coloco mais memória., com uma complicação a mais:

Se você estiver usando shadow passwords você deve colocar a letra x ou um asterisco no campo de senha do arquivo /etc/passwd para cada conta, dessa forma programas que não estejam cientes sobre shadow passwords não pensaram que a conta não possui senha.

Questão 6.8. Minha máquina roda muito lentamente quando eu executo GCC / X / ...

Talvez você esteja sem áreas de swap habilitadas. Você precisa habilitar a paginação (swapping) para que o Linux possa mover dados que os programas não estejam usando no momento para o disco, liberando memória para outros programas e dados. Se você não fizer isso, o Linux precisará se desfazer de porções de programas na memória, fazendo com que o Linux tenha que recarrega-los posteriormente.

Veja o Installation HOWTO e o Installation and Getting Started Guide [P2.1. Onde pego os HOWTOs e outra documentação?] para detalhes de como adicionar uma partição de swap ou um arquivo de swap; veja também P4.6. Minha área de paginação (swap) não funciona..

Alternativamente, você pode estar com muito pouca memória real. Se você tem menos memória que o usado por todos os programas que você estiver executando, o Linux irá usar o seu disco rígido e ficará terrivelmente lento. A solução neste caso é não rodar tantos programas ao mesmo tempo ou comprar mais memória. Você pode também liberar alguma memória compilando e usando um kernel com menos opções configuradas. Veja P7.6. Como atualizo/recompilo meu kernel? .

Você pode ver quanta memória e/ou swap você está usando com o comando free, ou digitando

```
cat /proc/meminfo
```

Se seu kernel está configurado com um ramdisk, isso provavelmente é um desperdício de espaço deixando o sistema lento. Use o LILO ou rdev para fazer o kernel não alocar a ramdisk (veja a documentação do LILO ou digite man rdev).

Questão 6.9. Eu só consigo me "logar" como root.

Provavelmente você tem problemas de permissão, ou tem um arquivo /etc/nologin.

No caso do último, coloque rm -f /etc/nologin no seu /etc/rc.local ou nos scripts /etc/rc.d/*.

Caso contrário verifique as permissões do seu shell, de quaisquer nomes de arquivos que apareçam em mensagens de erro, e também da hierarquia de diretórios que contém esses arquivos, inclusive o diretório raiz.

Questão 6.10. Minha tela está cheia de caracteres esquisitos ao invés de letras.

Você possivelmente mandou algum dado binário para a sua tela por engano. Digite echo '\033c' para consertar. Muitas distribuições de Linux contém um comando reset que faz isso.

Questão 6.11. Eu acabei com meu sistema e não consigo me logar para consertá-lo.

Inicialize o sistema com um (ou um par de) disquete de emergência, como por exemplo o par de discos boot e root do Slackware (no subdiretório install dos espelhos do Slackware) ou com o disquete de boot de instalação do MCC. Existem dois pacotes de criação de disquetes de emergência no sunsite.unc.edu em /pub/Linux/system/Recovery. Será melhor se você colocar o seu próprio kernel neles, dessa forma você não correrá o risco de ficar sem algum controlador de dispositivo, sistema de arquivos etc.

De algum prompt de shell monte seu disco rígido com algo tipo

```
mount -t ext2 /dev/hda1 /mnt
```

Assim seu sistema de arquivos estará acessível em /mnt e você poderá consertar o problema. Lembre-se de desmontar (umount /mnt) o seu disco rígido antes de reiniciar (volte ao diretório raiz antes ou ele dirá que está ocupado).

Questão 6.12. Descobri uma falha de segurança enorme no rm!

Não, você não descobriu. Obviamente você é novo no Unix e precisa ler um bom livro sobre ele para ver como as coisas funcionam. Pista: a habilidade de apagar arquivos no Unix depende de permissões de escrita no diretório onde eles estão.

Questão 6.13. lpr e/ou lpd não estão funcionando.

Veja Printing HOWTO [P2.1. Onde pego os HOWTOs e outra documentação?].

Questão 6.14. A hora de arquivos de partições msdos estão gravados incorretamente.

Existe um erro no programa clock (geralmente encontrado em /sbin) - ele conta incorretamente um ajuste de fuso horário, confundindo segundos com minutos ou coisa parecida. Pegue uma versão mais nova dele.

Questão 6.15. Como faço o LILO carregar o arquivo vmlinux?

Nos kernel versões 1.1.80 e posteriores a imagem comprimida do kernel, que é o que você deve dar ao LILO, foi movido para arch/i386/boot/zImage. O arquivo vmlinux no diretório root é um kernel descomprimido e você não deve tentar carregá-lo.

Questão 7.1. Como faço para rolar a tela para cima em modo texto?

Com o mapeamento de teclado (keymap) americano padrão, você pode usar a tecla Shift com PageUp ou PageDown (Nota: as teclas cinzas, não as brancas do teclado numérico!). Para outros mapeamentos de teclado, veja /usr/lib/keytables; você pode remapear as teclas de rolamento para cima (scroll up) e para baixo (scroll down) para o que você quiser --- por exemplo, para remapea-los para as teclas existentes em um teclado AT de 94 teclas.

Não é possível aumentar a quantidade de tela guardada, porque a forma que é implementado usa a memória de vídeo para guardar o texto rolado. Porém, você pode aumentar a quantidade de texto guardado em cada console virtual, reduzindo a quantidade deles -- veja o <linux/tty.h>.

Questão 7.2. Como troco de console virtual? Como os habilito?

Em modo texto, pressione Left Alt-F1 a Alt-F12 para selecionar os consoles tty1 a tty12; Right Alt-F1 vai para tty13 e assim em diante. Para mudar de console no X windows pressione Ctrl-Alt-F1 etc; Alt-F5 ou algo assim irá levá-lo de volta ao X.

Se você quer usar um console virtual para login normal, você precisa listar quais consoles virtuais possuem prompt de login em /etc/inittab. (Nota: o X precisa de pelo menos um console virtual para rodar).

Kernels anteriores ao 1.1.59 possuem um limite no número de consoles configurado em tempo de compilação, cujo default é 8. Veja NR_CONSOLES em linux/include/linux/tty.h. Kernels mais recentes alocam consoles dinamicamente, até um máximo de 63.

Questão 7.3. Como configuro o timezone (fuso-horário)?

Vá ao diretório /usr/lib/zoneinfo; pegue o pacote de timezone se você não possui esse diretório. Os fontes podem ser encontrados no sunsite.unc.edu em /pub/Linux/system/Admin/timesrc-1.2.tar.gz.

Faça um link simbólico com o nome localtime apontando para um dos arquivos nesse diretório (ou subdiretório), e um chamado posixrules apontando para localtime. Por exemplo:

```
ln -sf Brazil/West localtime
ln -sf localtime posixrules
```

Isso tomará feito imediatamente - experimente date.

Não tente usar a variável de ambiente TZ - deixe-a indefinida.

Você também deve certificar-se de que o relógio de seu kernel Linux está ajustado para o horário GMT correto - digite date -u e verifique se o horário universal correto é mostrado.

Questão 7.4. Que versão de Linux e que máquina estou usando?

Digite:

```
uname -a
```

Questão 7.5. Como posso habilitar ou desabilitar core dumps?

O Linux agora tem os corefiles desabilitados por default para todos os processos.

Você pode habilitar ou desabilitar os core dumps (gravação em disco da memória do processo, em alguns casos de erro fatal) com o comando ulimit no bash, limit no tcsh ou rlimit no ksh. Veja a página de manual para o seu shell para maios detalhes.

Esse comando afeta todos os programas executados por aquele shell (direta ou indiretamente) e não no sistema todo.

Se você quiser habilitar ou desabilitar core dumps para todos os processos por default mude <linux/sched.h> - veja a definição de INIT_TASK, e também dê uma olhada em <linux/resource.h>.

1.2.13 produzirá core dumps a.out [O que é esse tal de ELF?].

Questão 7.6. Como atualizo/recompilo meu kernel?

Leia o Kernel HOWTO ou o README que acompanha o pacote do kernel em ftp.cs.helsinki.fi, em /pub/Software/Linux/Kernel e espelhos de lá [P2.5. Onde eu pego material sobre Linux por FTP?]. Você pode já possuir uma versão do código fonte do kernel instalado em seu sistema, mas se você o obteve como parte de alguma distribuição do Linux, é provável que ele esteja desatualizado (o que não é problema se você só quer um kernel configurado sob medida, mas provavelmente é se você precisa fazer uma atualização.)

Lembre-se que para fazer o novo kernel ser carregado, você precisa rodar o LILO depois de copiar o kernel em sua partição raiz -- o Makefile de versões recentes do kernel possui um "target" zlilo especial que já faz isso; tente make zlilo.

Kernels com número de versão secundário ímpares (ie, 1.1.x, 1.3.x) são para

teste; kernels estáveis possuem número de versão pares (1.0.x, 1.2.x). Se você quiser experimentar algum kernel de teste, assine a lista de discussões linux-kernel [P2.8. Quais são as listas de discussão sobre Linux?].

Russel Nelson envia sumários de mudanças recentes no kernel para comp.os.linux.development , estes são arquivados em /pub/kchanges.

Questão 7.7. Posso ter mais de 3 portas seriais compartilhando interrupções?

Sim, mas você não poderá usar duas portas comuns que compartilham uma interrupção simultaneamente (sem usar alguns truques). Esta é uma limitação da arquitetura do barramento ISA.

Veja o Serial HOWTO para informações sobre possíveis soluções para este problema.

Questão 7.8. Como faço um disquete inicializável?

Faça um sistema de arquivos nele com os diretórios bin, etc, lib e dev -- tudo que você precisa. Instale um kernel nele e faça com que o LILO carregue-o pelo disquete (veja a documentação do LILO em lilo.u.*.pl).

Se você montou o kernel (ou fez o LILO dizer ao kernel) para ter um ramdisk do mesmo tamanho do disquete, o ramdisk será carregado em tempo de inicialização (boot) e será montado como raiz no lugar do disquete.

Veja o Bootdisk HOWTO.

Questão 7.9. Como remapeio meu teclado para português, francês etc.

Para kernels recentes, pegue /pub/Linux/system/Keyboards/kbd-0.90.tar.gz em sunsite.unc.edu. Certifique-se de pegar uma versão apropriada; você precisa do pacote de mapeamento de teclado específico para a versão do seu kernel. 0.90 deve funcionar com versões de kernel apartir de 1.0.

Para kernels mais antigos, você precisará editar o Makefile do kernel em /usr/src/linux.

Você pode encontrar mais informações úteis no Keystroke HOWTO, em sunsite.unc.edu em /pub/Linux/docs/HOWTO/Keystroke-HOWTO ou no Portuguese-HOWTO , no mesmo local.

Questão 7.10. Como faço para deixar o NUM LOCK ligado por default?

Use o programa setleds, por exemplo (em /etc/rc.local ou em um dos arquivos /etc/rc.d/*):

```
for t in 1 2 3 4 5 6 7 8
do
```

```
    setleds +num &lft; /dev/tty$t > /dev/null
  done
```

setleds é parte do pacote kbd (P7.8. Como faço um disquete inicializável?).

Alternativamente, modifique seu kernel. Você precisa fazer com que KBD_DEFLEDS seja definido para (1 << VC_NUMLOCK) ao compilar drivers/char/keyboard.c.

Questão 7.11. Como posso ter mais de 128Mb de swap?

Use várias partições ou arquivos de swap - o Linux suporta até 16 áreas de swap, cada um com até 128Mb.

Kernels muito velhos somente suportam áreas de swap com tamanhos de até 16Mb.

Questão 7.12. Como faço para obter acentuação no Linux?

Leia o Portuguese-HOWTO

Documentacao para outras linguas tambem podem ser uteis. Tente ver Danish-HOWTO.

Antonio Fernando C. M. Fo. afmoraes@if.usp.br esta organizando um HOWTO sobre o assunto. Ele ainda nao esta completo, mas ja possui muitas informacoes. Tente ver em ftp://linux.if.usp.br/pub/operating_systems/Linux/local. Note que esta localizacao do texto e' temporaria.

Questão 8.1. Como programo XYZ no Linux?

Leia os manuais ou um bom livro sobre Unix. As páginas de manual online (digite man man) são geralmente uma boa fonte de informações de referência sobre como usar um comando ou função exatamente.

Também existe bastante documentação em formato GNU Info, útil como tutorial. Rode Emacs e tecle Ctrl-h i, ou digite info info se você não tem, ou não gosta de Emacs. Note que o nodo libc do Emacs não descreve exatamente o libc do linux (que é mais parecido com um libc tradicional de Unix, não tendo algumas das esquisitices da GNU), mas é suficientemente semelhante para servir como tutorial de programação em C no Unix.

A última versão dos manuais online do Linux e uma coleção de documentação GNU Info, com várias outras informações sobre programação em Linux pode ser encontrada em sunsite.unc.edu em /pub/Linux/docs/man-pages.

Questão 8.2. O que é esse tal de ELF?

Veja o ELF HOWTO por Daniel Barlow, note que esse não é o arquivo move-to-elf, que é uma explicação de como atualizar para ELF manualmente.

Linux está migrando para um formato diferente de executáveis, arquivos objeto e bibliotecas de código objeto, conhecido por `ELF' (o formato antigo é chamado `a.out'). Este terá muitas vantagens, incluindo suporte melhorado para bibliotecas compartilhadas (shared libraries) e ligação dinâmica (dinamic linking).

Binários a.out e ELF podem coexistir em um sistema. Porém, como eles usam bibliotecas C compartilhadas diferentes, ambos tipos devem estar instaladas para isso.

Se você quer saber se seu sistema pode rodar binários ELF, procure /lib por um arquivo libc.so.5. Se este existir, ele provavelmente pode. Se você quer saber se a sua instalação relamtente é ELF você pode pegar um programa representativo, como ls, e rodar file nele:

```
-chiark:~> file /bin/ls
/bin/ls: Linux/i386 impure executable (OMAGIC) - stripped

valour:~> file /bin/ls
/bin/ls: ELF 32-bit LSB executable, Intel 80386, version 1, stripped
```

Existe um patch para fazer o 1.2.x compilar usando compiladores ELF, e produzir coredumps ELF, no tsx-11.mit.edu em /pub/packages/GCC. Você não precisa do patch para meramente rodar binários ELF. 1.3.x e posteriores não precisam do patch.

Questão 8.3. O que são arquivos .gz? E .tgz? E ...?

Arquivos .gz (e .z) foram comprimidos usando o GNU gzip. Você precisa usar o gunzip (que é um link simbólico para o comando gzip que vem com a maioria das instalações Linux) para descomprimir o arquivo.

.taz e .tz são tarfiles (feitos com o tar do Unix) comprimidos usando o o compress padrão do Unix.

.tgz (ou .tpz) é um tarfile comprimido com o gzip.

.lsm são entradas do Linux Software Map, sob forma de um arquivo texto curto. Detalhes sobre o LSM e o LSM em si estão disponíveis no subdiretório docs em sunsite.unc.edu.

.deb é o pacote binário Debian - o formato de pacote binário usado pela distribuição Debian GNU/Linux. Ele é manipulado com dpkg e dpkg-deb (disponíveis em sustemas Debian e em ftp.debian.org).

.rpm é o pacote Red Hat RPM, que é usado na distribuição Red Hat. Eles podem ser encontrados em ftp.redhat.com.

O comando file pode, na maioria dos casos, dizer o que é um arquivo.

Se o gzip reclamar quando você tenta descomprimir um arquivo comprimido por ele, você provavelmente transferiu o arquivo em modo ASCII por engano. Você deve fazer a transferência (download) da maioria das coisas em modo binário - lembre-se de digitar binary como um comando no FTP antes de usar o get para pegar o arquivo.

Questão 8.4. O que significa VFS?

Virtual File System. É uma camada de abstração entre o usuário e os sistemas de arquivos reais como ext2, minix e msdos. Entre outras coisas, sua função é esvaziar o buffer de leitura quando detecta uma mudança de discos na unidade de disco flexível:

VFS: Disk change detected on device 2/0

Questão 8.5. O que é um BogoMip?

`BogoMips' é uma contração de `Bogus MIPS'. MIPS significa (dependendo de por quem você ouviu falar) Milhões de Instruções Por Segundo, ou Meaningless Indication of Processos Speed.

O número mostrado em tempo de inicialização é o resultado de uma calibragem de tempo do kernel, usado para loops de espera bastante curtos por alguns controladores de dispositivo.

Como um guia bastante grosseiro, o BogoMips será aproximadamente:

```
386SX           clock * 0.14
386DX           clock * 0.18
486Cyrix/IBM    clock * 0.33
486SX/DX/DX2    clock * 0.50
586             clock * 0.39
```

Se o número que você vê é sensivelmente inferior ao acima, você pode estar com o botão do Turbo ou velocidade da CPU configurados incorretamente, ou ter algum tipo de problema de cache [como descrito em P6.5. Linux fica muito lento quando coloco mais memória.].

Para valores encontrados em outros chips mais raros veja o BogoMips Mini-HOWTO, no sunsite.unc.edu em /pub/Linux/docs/howto/mini/BogoMips.

Questão 8.6. O que é o Linux Journal e onde posso consegui-lo?

Linux Journal é uma revista mensal (impressa em papel) que é disponível em bancas e por assinaturas mundialmente. Mande email para linux@ssc.com para detalhes. Eles estão na WWW em http://www.ssc.com/.

Questão 8.7. Quantas pessoas usam Linux?

Linux é disponível gratuitamente, e ninguém é obrigado a registrar suas cópias em nenhuma autoridade central, portanto isso é difícil de se saber. Muitas empresas sobrevivem agora somente da venda e suporte ao Linux, e relativamente muito poucos usuários de Linux usam esses serviços. Os grupos de discussão sobre Linux são alguns dos mais lidos na Rede, assim o número deve estar na casa das centenas de milhares, mas números exatos são difíceis de se dizer.

Contudo, uma brava alma, Harald T. Alvestrand Harald.T.Alvestrand@uninett.no, decidiu tentar e pede que se você usa Linux, mande uma mensagem para linux-counter@uninett.no com uma das seguintes linhas de subject:`I use Linux at home', `I use Linux at work', ou `I use Linux at home and at work'. Ele também aceitará registros de terceiros - pergunte a ele por detalhes.

Alternativamente, você pode registrar-se usando formulários WWW em http://domen.uninett.no/~hta/linux/counter.html.

Ele envia suas contagens para comp.os.linux.misc todo mês; você também pode olhar em aun.uninett.no em /pub/misc/linux-counter ou na página WWW acima.

Questão 8.8. Como se pronuncia Linux?

Este é um debate religioso, claro!

Se você quiser ouvir o próprio Linus pronuncia-lo pegue english.au ou swedish.au em ftp.funet.fi (em /pub/OS/Linux/PEOPLE/Linus/SillySounds). Se você tiver uma placa de som ou o driver de áudio para PC-speaker, você poderá ouvi-los digitando

```
cat english.au >/dev/audio
```

A diferença não é na pronúncia de Linux mas na língua que Linus usa para dizer hello. A versão em inglês parodiada muito bem por Jin Choi as "Hi, my name is Leenoos Torvahlds and I pronounce Leenooks as Leenooks."

Para o benefício daqueles que não têm o equipamento nem a inclinação: Linus pronuncia Linux aproximadamente como se lê em português: Linux.

Questão 9.1. "Unknown terminal type linux" e semelhantes

Em versões 1.3.x antigas do kernel, o tipo de terminal do console default mudou de console para linux. Você deve editar o /etc/termcap e mudar a linha:

```
console|con80x25:\
```

para

```
linux|console|con80x25:\
```

(Se houver um dumb a mais nesta linha, ela deverá ser removida.)

Para fazer o editor funcionar, você provavelmente terá que digitar TERM=console (para o bash e ksh) ou setenv TERM console (csh, tcsh) antes.

Alguns programas usam /usr/lib/terminfo em vez do /etc/termcap. Para esses programas você deve atualizar seu terminfo, que é parte do ncurses.

Questão 9.2. Durante a "linkagem" aparecem Undefined symbol _mcount

Isso geralmente se deve a uma má interação entre um defeito do SLS e as notas de instalação da biblioteca C. Seu libc.a foi trocado pela biblioteca de testes. Você deve remover libc.a, libg.a e libc_p.a e reinstalar as bibliotecas (seguindo as notas de instalação delas).

Questão 9.3. lp1 on fire

Esta é um mensagem tradicional/de brincadeira indicando que algum tipo de erro está sendo reportado pela sua impressora, mas o status do erro é inválido. Pode ser que você tem algum tipo de conflito de I/O ou IRQ - verifique a configuração de suas placas. Algumas pessoas dizem que recebem esta mensagem quando suas impressoras estão desligadas. Provavelmente ela não está realmente em chamas...

Em kernels mais recentes a mensagem foi mudada para lp1 reported invalid error status (on fire, eh?).

Questão 9.4. INET: Warning: old style ioctl(IP_SET_DEV) called!

Você está tentando usar os utilitários de configuração de rede antigos; os novos podem ser encontrados em ftp.linux.org.uk em /pub/linux/Networking/PROGRAMS/NetTools (somente fontes, acho).

Note que estes não podem ser usados como os antigos; veja o NET-2 HOWTO para instruções de como configurar redes corretamente.

Questão 9.5. ld: unrecognized option '-m486'

Você possui uma versão antiga do ld. Instale um pacote binutils mais recente -- ele conterá um ld mais recente. Procure por binutils-2.6.0.2.bin.tar.gz em tsx-11.mit.edu em /pub/linux/packages/GCC.

Questão 9.6. GCC diz Internal compiler error

Se o erro é reprodutível (i.e, ele sempre ocorre no mesmo local do mesmo arquivo --- mesmo após reinicializar e tentar novamente, usando um kernel estável) você descobriu um erro no GCC. Veja a documentação do GCC em

formato Info (digite Control-h i no Emacs, e selecione GCC no menu) para detalhes de como reportar isso -- no entanto, certifique-se de que você possui a última versão do GCC.

Note que este provavelmente não é um erro específico do Linux; a não ser que você esteja compilando um programa que muitos outros usuários Linux compilam, você não deve enviar seu aviso sobre o erro para os grupos comp.os.linux.

Se o erro for irreprodutível você provavelmente está experimentando corrupção de memória --- veja P9.7 make diz Error 139.

Questão 9.7. make diz Error 139

Seu compilador (gcc) fez um core dump. Você possivelmente tem uma versão velha, com erros ou corrompida do GCC -- pegue a última versão. Você tembém pode estar com pouco espaço de swap -- veja P6.8. Minha máquina roda muito lentamente quando eu executo GCC / X / ... para mais informações.

Se isto não consertar o problema você está provavelmente tendo problemas de corrupção de memória ou disco. Verifique se a velocidade do clock, wait states etc. de seus SIMMs e cache estão corretos (manuais de hardware algumas vezes estão incorretos). Se estiverem, você pode estar com SIMMs defeituosos ou com velocidades conflitantes ou placa-mãe ou controladora ou disco defeituosos.

Linux, como qualquer Unix, é um excelente testador de memória --- muito melhor que programas de teste de memória baseados em DOS.

Alguns clones de coprocessadores aritmético x87 podem causar problemas; tente compilar um kernel com emulação de processamento aritmético (Kernel math emulation) habilitado; você precisará passar o comando no387 para o kernel, no prompt do LILO, para forçar o kernel a usá-lo ou o kernel não usará o código de emulação.

Mais informações sobre este problema esté disponível no WWW em http://einstein.et.tudelft.nl/~wolff/sig11/.

Questão 9.8. shell-init: permission denied quando me logo.

Seu diretório raiz e todos os diretórios até o seu diretório home devem estar com permissão de leitura e execução para todos. Veja a página de manual do chmod ou um livro de Unix para ver como consertar o problema.

Questão 9.9. No utmp entry. You must exec ... quando me logo.

Seu /var/run/utmp está corrompido. Você deve ter

```
> /var/run/utmp
```

em seu /etc/rc.local ou /etc/rc.d/*. Veja Eu acabei com meu sistema e não consigo me logar para consertá-lo. para ver como poder fazer isso. Note que utmp pode também ser encontrado em /var/adm/utmp ou /etc/utmp em alguns sistemas mais antigos.

Questão 9.10. Warning - bdflush not running

Kernels modernos usam estratégias melhores para escrever blocos de disco em cache. Em adição às mudanças no kernel, isto envolve a troca do programa update, que costumava escrever tudo a cada 30 segundos, por um (na verdade dois) daemon mais sutil, conhecido por bdflush.

Pegue o bdflush-n.n.tar.gz do mesmo local que os fontes do kernel [P7.6. Como atualizo/recompilo meu kernel?], compile-o e instale-o; ele deve ser iniciado antes da checagem usual dos sistemas de arquivos em tempo de inicialização. Ele funcionará sem problemas com kernels mais antigos, portanto não há motivos para manter o update antigo.

Questão 9.11. Warning: obsolete routing request made.

Não se preocupe com isto; isto só quer dizer que a versão do route que você tem é um pouco desatualizado em relação ao kernel. Você pode se livrar da mensagem pegando uma versão mais recente do route do mesmo local que os fontes do kernel [P7.6. Como atualizo/recompilo meu kernel?].

Questão 9.12. EXT2-fs: warning: mounting unchecked filesystem

Você deve rodar e2fsck (ou fsck -t ext2 se você tem o programa fsck) com a opção -a para limpar o flag de `dirty', e depois desmontar a partição durante cada finalização (shutdown) do sistema.

A forma mais fácil de fazer isto é pegar a última versão dos comandos fsck, umount e shutdown, disponíveis no pacote util-linux de Rik Faith [P2.5. Onde eu pego material sobre Linux por FTP?]. Você deve certificar-se de que os scripts /etc/rc* usam-nos corretamente.

Nota: não tente checar um sistema de arquivos que está montado como read-write - isso inclui a partição raiz se você não vir

VFS: mounted root ... read-only

durante a inicialização. Você deve fazer com que a partição raiz seja montada somente para leitura, checá-la se necessário e então remontá-la como read-write. Leia a documentação que acompanha com o util-linux para ver como fazer isso.

Note que você deve especificar a opção -n para mount para fazer com que ele não tente atualizar o /etc/mtab, caso contrário ele falhará, pois o sistema de arquivos raiz (onde está /etc/mtab) ainda é somente para leitura.

Questão 9.13. EXT2-fs warning: maximal count reached

Esta mensagem é emitida pelo kernel quando ele monta um sistema de arquivos marcado como limpo, mas cujo contador de "montagens desde a última checagem" atingiu um valor pré-definido. A solução é pegar a última versão dos utilitários ext2fs (e2fsprogs-0.5b.tar.gz no momento da escrita disto) dos sites usuais [P2.5. Onde eu pego material sobre Linux por FTP?].

O número máximo de montagens pode ser examinado e mudado com o programa tune2fs deste pacote.

Questão 9.14. EXT2-fs warning: checktime reached

Kernels 1.0 em diante suportam a checagem de sistemas de arquivos baseados no tempo decorrido desde a última checagem assim como no número de montagens. Pegue a última versão dos utilitários ext2fs (e2fsprogs-0.5b.tar.gz no momento da escrita disto) dos sites usuais [P2.5. Onde eu pego material sobre Linux por FTP?].

Questão 9.15. df diz Cannot read table of mounted filesystems

Provavelmente há algo de errado com seus arquivos /etc/mtab ou /etc/fstab. Se você possuir uma versão razoavelmente recente do mount, /etc/mtab deve ser esvaziado ou apagado em tempo de inicialização (em /etc/rc.local ou /etc/rc.d/*), usando algo como

```
rm -f /etc/mtab*
```

Algumas versões de SLS têm uma entrada para a partição raiz em /etc/mtab feito no /etc/rc* usando rdev. Isso é incorreto: versões recentes de mount fazem isso automaticamente.

Outras versões do SLS têm uma linha em /etc/fstab que se parece com:

```
/dev/sdb1   /root   ext2   defaults
```

Isso é errado. /root deve ser simplesmente /.

Questão 9.16. fdisk says Partition X has different physical /logical...

Se o número da partição (X, na mensagem acima) for 1 esse é o mesmo problema que fdisk: Parttion 1 does not start on cylinder boundary.

Se a partição começa ou termina em um cilindro com número maior que 1024, isso é devido ao fato das informações do DOS sobre geometria do disco não suportarem números de cilindro com mais de 10 bits. Leia P4.1. Como posso fazer o Linux funcionar com meu disco "grande" de mais de 1024 cilindros lógicos?.

Questão 9.17. fdisk: Partition 1 does not start on cylinder boundary

A versão do fdisk que vem com muitos sistemas Linux cria partições que falham em seus próprios testes de consistência. Infelizmente, se você já instalou seu sistema não há muito o que se fazer sobre isso, além de copiar os dados da partição, removê-lo, reconstruí-lo e copiar os dados de volta.

Você pode evitar o problema pegando a última versão de fdisk, do pacote util-linux de Rik Faith (disponível em todos os bons sites de FTP). Alternativamente, se você estiver criando uma nova partição 1 que começa no primeiro cilindro, você pode fazer o seguinte para obter uma partição que o fdisk goste:

1. Crie a partição 1 normalmente. Uma listagem p produzirá uma reclamação sobre inconsistência.
2. Digite u para mudar a unidade de medida para setores e digite p novamente. Copie o número da coluna "End".
3. Remova a partição 1.
4. Ainda com a unidade de medida em setores recrie a partição 1. Faça com que o primeiro setor coincida com o número de setores por trilha. Este é o número de setor na primeira linha da saída de p. Ajuste o último setor para o valor indicado no passo 2.
5. Digite u para mudar a unidade de volta para cilindros e continue com as outras partições.

Ignore as mensagens sobre setores não alocados - eles referem-se aos setores na primeira trilha além do MBR, que não é usado se você começar a primeira partição na trilha 2.

Questão 9.18. fdisk says partition n has an odd number of sectors

O esquema de particionamento de disco do PC funciona com setores de 512 bytes, mas Linux usa blocos de 1K. Se você tiver uma partição com um número ímpar de setores o último setor não é utilizado. Ignore a mensagem.

Questão 9.19. mtools diz cannot initialise drive XYZ

Isso significa que mtools está tendo problemas em acessar o drive. Isso pode ser causado por várias coisas.

Freqüentemente isso ocorre devido às permissões nos dispositivos das unidades de disco flexível (/dev/fd0* e /dev/fd1*) estarem incorretas --- o usuário rodando mtools deve ter acesso apropriado. Veja a página de manual do chmod para detalhes.

A maioria das versões do mtools distribuídos com sistemas Linux (não a versão padrão GNU) usa o conteúdo do arquivo /etc/mtools para ver quais dispositivos e densidades usar, ao invés de ter essas informações compilados no programa. Erros nesse arquivo podem causar problemas. Freqüentemente não há documentação sobre isso --- produtores de distribuições por favor notem

que isso é mau.

Para a forma mais fácil de acessar seus arquivos DOS (especialmente aqueles em partições de disco rígido) veja P3.2. Como eu acesso arquivos na minha partição DOS ou em disquetes? . Nota - você jamais deve usar mtools para acessar arquivos em uma partição ou disquete msdosfs montado!

Questão 9.20. Memory tight no começo da inicialização

Isso significa que você tem um kernel muito grande, o que significa que o Linux tem que fazer alguma mágica de gerenciamento de memória extra para poder inicializar-se apartir do BIOS. Isso não é relacionado com a quantidade de memória física em sua máquina. Ignore a mensagem ou compile um kernel contendo somente os drivers e funções que você precisa P7.6. Como atualizo/recompilo meu kernel?

Questão 9.21. You don't exist. Go away.

Isso não é uma infecção viral :-). Isso vem de vários programas como write, talk e wall, se o seu uid efetivo ao executar o programa não corresponde a um usuário válido (provavelmente devido a uma corrupção no /etc/passwd), ou se a sessão (pseudoterminal, especificamente) que você está usando não estiver registrado no arquivo utmp (provavelmente porque você o invocou de um jeito engraçado)

Questão 10.1. O Linux suporta X Windows?

Sim. Linux usa XFree86 (a versão atual é 3.1.2, que é baseado no X11R6). Você deve ter uma placa de vídeo suportado pelo XFree86. Veja o Linux XFree96 HOWTO para maiores detalhes.

Atualmente, a maioria das distribuições Linux vêm com uma instalação X.

Você tambem pode instala-lo por conta propria, com /pub/Linux/X11/Xfree86-* em sunsite.unc.edu e seus sites espelho. Leia o XFree86 HOWTO para informações sobre instalação.

Questão 10.2. Onde posso conseguir um XF86Config para meu sistema?

Leia o Linux XFree86 HOWTO.

Você deverá fazer seu próprio arquivo XF86Config, porque ele depende da combinação exata de placa de vídeo e monitor que você tem. Isso não é difícil de fazer -- leia as instruções que vem com o XFree86, em /usr/X11R6/lib/X11/etc. O arquivo que você mais precisa olhar é provavelmente o README.Config.

Você pode usar o programa xf86config para agilizar o trabalho. Note que ConfigXF86 é obsoleto e não deve ser usado.

Por favor, não peça por arquivos XF86Config em comp.os.linux.x nem responda a tais pedidos.

Questão 10.3. Logins pelo "xterm" aparecem de forma estranha em who, finger etc.

O xterm que acompanha o XFree86 2.1 e anteriores não entende corretamente o formato que o Linux usa para o arquivo /var/adm/utmp, onde o sistema guarda registros de quem está no sistema. Portanto ele não interpreta todas as informações corretamente.

XFree96 3.1 conserta esse problema.

Questão 10.4. Não consigo fazer o X Windows funcionar

Leia o XFree86 HOWTO - note a seção de perguntas e respostas.

Tente ler comp.windows.x.i386unix - especificamente leia o FAQ daquele grupo.

Por favor não envie mensagens sobre X Windows ou XFree86 para comp.os.linux.x a não ser que sejam específicos sobre Linux.

Questão 10.5. O que faço para o X funcionar com a minha TGUI9440, Diamond Stealth...?

Suporte para placas TGUI9xxx e outras está sendo desenvolvido. Se você quiser experimentar uma versão beta, dê uma olhada em http://www.xfree86.org.

Placas Diamond mais antigas não são suportadas, mas você pode fazê-las funcioná-las em sua máquina com algum trabalho. Veja Diamond.FAQ em ftp://sunsite.unc.edu/pub/Linux/X11/. Placas mais recentes possuem suporte. Obtenha mais informações em http://www.diamondmm.com/linux.html

Mais informações sobre suporte a placas de vídeo no Hardware HOWTO ou para o caso do XFree86 em http://www.xfree96.org

Questão 11.1. Emacs só faz core dump

Você provavelmente possui uma versão do Emacs que foi compilado para funcionar com o X11; ele precisa das bibliotecas X11 para funcionar. Se você estiver usando Slackware, você pode mudar o link simbólico /usr/bin/emacs para apontar para emacs-19.29-no-x11 (veja man ln). Ele está no arquivo emac_nox.tgz do disco E6 do Slackware.

Questão 11.2. fdisk diz cannot use nnn sectors of this partition

Originalmente o Linux só suportava o sistema de arquivos do Minix, que não pode usar mais de 64Mb por partição. Essa limitação não existe em sistemas de arquivos mais avançados, como o ext2fs (a segunda versão do Sistema de Arquivos Extendido, o sistema de arquivos padrão do Linux).

Se você pretende usar o ext2fs, você pode ignorar a mensagem.

Questão 11.3. GCC algumas vezes usa quantidades enormes de memória virtual e quebra

Versões antigas do GCC tinham um erro que os fazia usar muita memória se você tentasse compilar um programa com uma tabela de dados estático nele.

Você pode tanto atualizar seu GCC para pelo menos a versão 2.5, ou adicionar mais espaço de swap se necessário e simplesmente suportá-lo; no final ele irá funcionar.

Questão 11.4. Meu teclado fica todo esquisito quando troco de VCs.

Este é um erro em kernels de versões anteriores a 0.99pl14-alpha-n. Algumas vezes Linux se perde quanto a quais teclas modificadoras (Shift, Alt, Control etc.) estão pressionadas ou não, e acha que um ou outro estão pressionadas, quando não estão. A solução é pressionar e soltar cada uma das teclas modificadoras (sem pressionar outras teclas junto) --- isso fará com que Linux saiba do estado em que o teclado realmente está.

Este problema ocorre freqüentemente ao se sair do X Windows; ele pode ser evitado algumas vezes ao largar rapidamente as teclas Ctrl e Alt depois de se apertar a tecla de função do VC para que você está mudando.

Questão 12.1. Vocês ainda não responderão à minha pergunta.

Por favor leia toda esta resposta antes de postar. Sei que ela é um pouco comprida, mas você pode estar a ponto de fazer papel de bobo na frente de 50000 pessoas e gastar centenas de horas de seus tempos. Você não acha que vale a pena gastar algum tempo seu lendo e seguindo estas instruções?

Se você achar que alguma resposta está incompleta ou incorreta, por favor envie e-mail para Ian Jackson em ijackson@gnu.ai.mit.edu ou Alfredo Kojima em kojima@inf.ufrgs.br

Leia os livros apropriados do Linux Documentation Project - veja P2.1. Onde pego os HOWTOs e outra documentação?.

Se você é um novato no Unix leia o FAQ para comp.unix.questions, e aqueles para quaisquer grupos comp.unix.* que podem ser relevantes.

Linux é um clone de Unix, portanto quase tudo que você ler lá pode se aplicar a Linux. Esses FAQs, assim como todos os FAQs, podem ser encontrados

em rtfm.mit.edu em /pub/usenet/news.answers (o mail-server@rtfm.mit.edu pode mandar esses arquivos para os que não tem acesso FTP). Esses são espelhos dos arquivos de FAQ rtfm em vários sites - cheque o Introduction to *.answers na USENET ou veja news-answers/introduction no diretório acima.

Cheque o HOWTO relevante para o assunto em questão, se existe um, ou um documento sub-FAQ no estilo antigo apropriado. Veja os sites FTP.

Tente experimentar -- esse é a melhor forma de se conhecer Unix e Linux.

Leia a documentação. Veja as páginas de manual (digite man man se você não os conhece. Tente man -k <assunto> : algumas vezes isso lista páginas de manual relevantes e úteis).

Verifique a documentação Info (digite Control-h i, no Emacs) --- nota: isso não é só para o Emacs; por exemplo, a documentação do GCC está lá também.

Freqüentemente haverá um arquivo README junto com o pacote dando instruções de uso e instalação.

Certifique-se de que você não tem cópias velhas ou corrompidas do programas em questão. Se possível, pegue o novamente e reinstale-o -- talvez você tenha cometido um engano da primeira vez.

Leia comp.os.linux.announce. Muitas vezes ele contém informações importantes para todos usuários Linux.

Questões sobre X Windows gerais pertencem a comp.windows.x.i386unix, não em comp.os.linux.x. Mas leia o grupo antes (incluindo o FAQ), antes de você postar!

Somente depois de ter feito tudo isso e você ainda estiver encalhado você deve enviar sua mensagem para o grupo comp.os.linux.* ou lista de discussões apropriada. Leia antes a próxima questão: P12.2 O que colocar num pedido de ajuda.

Questão 12.2. O que colocar em um pedido de ajuda.

Por favor leia cuidadosamente o seguinte conselho sobre como escrever sua mensagem. Ao segui-los você aumentará as chances de um expert e/ou companheiro usuário lendo sua mensagem tenha informações suficientes e motivação para responder.

Este conselho se aplica tanto para mensagens para grupos/listas de discussão pedindo ajuda quanto para email mandado para experts e companheiros usuários.

Certifique-se de dar detalhes completos do problema, incluindo:

Com qual programa exatamente que você está tendo problemas. Inclua o número da versão, se conhecido, e diga onde você o pegou. Muitos comandos padrão dizem suas versões ao serem invocados com a opção

--version.
Qual distribuição de Linux você está usando (MCC, Slackware, Debian etc.) e qual versão dessa distribuição.
Uma transcrição exata e completa do texto de quaisquer mensagens de erro mostradas.
Exatamente que efeitos você esperava e exatamente que efeitos você observou. Uma transcrição de uma sessão exemplo é uma boa maneira de se fazer isso.
O conteúdo de quaisquer arquivos de configuração usados pelo programa em questão e quaisquer programas relacionados.
Quais versões do kernel e biblioteca dinâmica você está usando. A versão do kernel pode ser verificada com uname -a, e o da biblioteca dinâmica com ls -l /lib/libc.so.4.
Se parecer apropriado, detalhes de qual hardware você está usando.

Você corre pouco risco de deixar sua mensagem comprida demais a não ser que inclua grandes pedaços de código fonte ou arquivos com uuencode, portanto não se preocupe em dar informações de mais.

Use uma linha de Subject clara e detalhada. Não coloque coisas como 'não funciona', 'Linux', 'ajuda' ou 'pergunta' nele --- nos já sabemos disso!
Guarde o espaço para o nome do programa, um pedaço da mensagem de erro, sumário do comportamento incomum etc.

Se você estiver reportando uma mensagem `unable to handle kernel paging request', siga as instruções do README no código fonte do kernel, para transformar os números em algo mais significativo. Se você não fizer isso, ninguém que ler sua mensagem poderá fazer isso por você, já que o mapeamento dos números para nomes de funções varia de um kernel para outro.

Coloque um parágrafo de sumário no topo de sua mensagem.

No fim de sua mensagem, peça por respostas por email particular e diga que você enviará um sumário. Após alguns dias envie um sumário. Não faça simples concatenações das respostas que você recebeu --- sumarise. Colocar a palavra SUMÁRIO na linha de Subject de sua mensagem é também uma boa idéia. Considere em submeter seu sumário para comp.os.linux.announce.

Se for postar a mensagem para grupos USENET, certifique-se de que sua mensagem não tem uma linha References inapropriada. Isso marca seu artigo como parte da discussão referida, o que irá seguidamente fazer com que ele não seja lido por leitores junto com o resto de uma discussão desinteressante.

Se você pode querer dizer em sua mensagem que você já leu este FAQ e os HOWTOs apropriados.

Lembre-se de que você não deve postar mensagens enviadas particularmente a você sem a autorização do autor.

Questão 12.3. Quero mandar e-mail a alguém sobre meu problema.

Tente encontrar o autor ou desenvolvedor do programa ou componente que está lhe causando dificuldades. Se você tiver um local de contato para a distribuição Linux que você está usando, use-o.

Coloque tudo em seu email que você colocaria em uma mensagem pedindo por ajuda.

Finalmente, lembre-se de que apesar do fato da maioria da comunidade Linux ser bastante prestativa e responder a perguntas por email, você estará pedindo ajuda a um voluntário não pago, portanto você não tem o direito de esperar uma resposta.

Questão 13.1. Comentários são bem-vindos

Por favor envie seus comentários sobre este FAQ.

Sugestões de perguntas para este FAQ são aceitas em qualquer formato; quaisquer contribuições, comentários e correções serão muito bem vindas.

Por favor, mande-as para kojima@inf.ufrgs.br. Contribuições que se referem à versão original em inglês deste FAQ ou que podem ser úteis a outros usuários em geral (não só para os brasileiros) podem ser mandadas tanto para kojima@inf.ufrgs.br como para o autor da versão original, ijackson@gnu.ai.mit.edu .

Ao se referir a alguma(s) pergunta(s) faça-a pelo cabeçalho da(s) pergunta(s), ao invés dos números, já que a numeração não é fixa.

Questão 13.2. Formatos em que este FAQ é disponível

A versão original em inglês deste documento é disponível em arquivos texto ASCII, documento Info do Emacs, uma página HTML na World Wide Web, PostScript e como mensagem do USENET news.

As versões em ASCII, Info do Emacs, HTML e a da USENET e um arquivo para o formatador de textos Lout (de onde a versão PostScript é produzido) são gerados automaticamente por um script Perl e um arquivo no Bizarre Format with No Name.

Os arquivos gerados linux-faq.ascii, .inf e .ps um arquivo tar linux-faq.source.tar.gz, contendo o arquivo fonte BFNN e os scripts de conversão em Perl está disponível nos diretórios docs dos sítios FTP.

A versão HTML deste FAQ em inglês é disponível como http://www.cl.cam.ac.uk/users/iwj10/linux-faq/index.html e é espelhado em www.li.org e em outros lugares.

A versão USENET é postado regularmente para comp.os.linux.announce, comp.os.linux.answers, comp.answers e news.answers.

A versão brasileira do FAQ (este que você está lendo) é disponível em ASCII

puro e HTML, além de ser postado à Linux-BR periodicamente e estar disponível no servidor de listas da Linux-BR.

Você pode encontrar um arquivo contendo todas as versões e outro somente com a versão ASCII em http://www.inf.ufrgs.br/~kojima/linux/pack/.

Questão 13.3. Autores e agradecimentos

A versão original do Linux FAQ, em que esta versão foi baseada, foi compilada por Ian Jackson ijackson@gnu.ai.mit.edu, com ajuda e comentários de muitas outras pessoas.

Agradecimentos especiais são devidos a Matt Welsh, que moderou o comp.os.linux.announce e comp.os.linux.answers, coordenava os HOWTOs e escreveu uma parte substancial deles, a Greg Hankins, que coordena os HOWTOs agora, a Lars Wirzenius que modera comp.os.linux.announce, e a Marc-Michel Corsini, que escreveu o Linux FAQ original.

Agradecimentos também para as muitas pessoas que mandaram seus valiosos comentários e sugestões, numerosos demais para serem listados.

E por último, mas não menos importante, obrigado a Linus Torvalds e os outros contribuidores do Linux, por dar a nós o que escrever.

Esta versão brasileira do FAQ foi traduzida e extendida por Alfredo K. Kojima,kojima@inf.ufrgs.br, Marlon Borba,trfsinf@eu.ansp.br.

Questão 13.4. Copyright e outras informações legais

Este é o copyright do documento em que este FAQ se baseia:

Note that this document is provided as is. The information in it is not warranted to be correct; you use it at your own risk.

Linux Frequently Asked Questions with Answers is Copyright 1994, 1995, 1996 by Ian Jackson ijackson@gnu.ai.mit.edu.

It may be reproduced and distributed in its entirety (including this authorship, copyright and permission notice) provided that either:

- the distribution is not commercial (commercial means any situation in which you benefit financially - directly or indirectly such as by inclusion in a publication which carries commercial advertising), or
- the distribution is in machine-readable form (ie, a form intended to be easily processed by a computer).

Note that this restriction is not intended to prohibit charging for the service of printing or copying a document supplied by your customer.

Any distribution of a partial copy or extract, a translation or a derivative work must be approved by me before distribution. Email me - I'll probably be

C - Copyrights do Linux e Esquema de numeração das versões do Linux

O Linux não é um software de domínio público, mas sob a GNU Public License (veja apêndice), o código fonte do Linux pode permanecer livremente disponível. As pessoas podem cobrar pela cópia do Linux, se desejarem, desde que, com isso, não limite a distribuição do Linux.

Muitas pessoas, pelo mundo, tem trabalhado conjuntamente para criar o Linux, sob a direção de Linus Torvalds, o autor original, e cada uma mantém os direitos de copyright sobre o código que ele, ou ela, tenham escrito.

A versão 1.0 do Linux foi liberada no dia 14 de março de 1994 e, depois disso, foram feitas numerosas versões novas. Ainda existem bugs no sistema (como em qualquer sistema operacional), e novos bugs vem sendo descobertos e ajustados no decorrer do tempo. O Linux segue o modelo de desenvolvimento aberto e, por isso, a cada nova versão liberada ao público, quer sim, quer não, ele é considerado um "produto de qualidade". Contudo, para dizer às pessoas se elas estão obtendo uma versão estável ou não, o esquema a seguir foi implementado:

Versões r.x.y, onde x é um número par, são versões estáveis, e, enquanto o y é incrementado, apenas reparos de bugs são efetuados. Assim da versão 2.0.2 para a 2.0.3, houve apenas reparo de bugs, sem nenhuma característica nova.

Versões r.x.y, onde x é um número impar, são versões beta destinadas apenas a desenvolvedores, podem ser instáveis e falhar, e estarão sendo colocadas novas característica o tempo todo. De tempos em tempos, com o atual desenvolvimento do kernel sendo considerado "estável", x é trocado para um número par, e o desenvolvimento continua com uma nova versão (x impar).

D - Contrato de licença (em Inglês)

O texto a seguir pode ser encontrado em: **http://www.cs.helsinki.fi/linux/gpl.html**

NOTE!
This copyright does not cover user programs that use kernel services by normal system calls - this is merely considered normal use of the kernel, and does not fall under the heading of "derived work". Also note that the GPL below is copyrighted by the Free Software Foundation, but the instance of code that it refers to (the Linux kernel) is copyrighted by me and others who actually wrote it.

Linus Torvalds

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBCLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Bibliografia

Servidores www

http://www.openline.com.br/linux-br/ - Home Page do Linux no Brasil
http://www.br.freebsd.org/linux-br/index.html - Home Page do Linux em português
http://www.inf.ufrgs.br/~kojima/linux/faq-linux.html - Perguntas frequentemente colocadas na linux-br (FAQ)
http://www.linux.org - Home page da Linux Organization, site oficial do Linux
http://www.suncite.unc.edu/mdw/welcome.html - Página do Projeto de Documentação do Linux.
http://www.linux.if.usp.br - Tudo para linux em português/inglês site da USP.
http://www.infor.es/LuCAS - Projeto Lucas - informações do Linux em espanhol.
http://www.cl.com.ac.uk/users/wj10/linux-faq - Pergunta mais frequentes do Linux em inglês.
http://sunsite.unc.edu/mdw/linux.html - Site com tudo sobre Linux em Inglês

Servidores de FTP

ftp.iis.com.br - diversos arquivos da internet.
ftp.versatec.com - contém diversos softwares para Linux.
ftp.ibp.fr : /pub/linux (França)
ftp.cc.gatech.edu : /pub/linux (EUA - sudeste: Suranet)
ftp.cdrom.com : /pub/linux (EUA)
ftp.informatik.tu-muenchen.de : /pub/comp/os/linux (Alemanha)
ftp.ibr.cs.tu-bs.de : /pub/os/linux (Alemanha)
ftp.dfv.rwth-aachen.de : /pub/linux (Alemanha)
ftp.informatik.rwth-aachen.de : /pub/Linux (Alemanha)
ftp.cc.monash.edu.au : /pub/linux (Austrália)
ftp.dstc.edu.au : /pub/Linux (Austrália: Queensland)
ftp.sun.ac.za : /pub/linux (África do Sul)
ftp.inf.utfsm.cl : /pub/Linux (Chile)
ftp.zel.fer.hr : /pub/Linux (Croácia)
linux.if.usp.br : /pub/mirror/sunsite.unc.edu/Linux além de outros em
/pub/mirror com a distribuicao Debian.
lcmi.ufsc.br : /pub/diversos/linux (Brasil : Santa Catarina) Slackware
cesar.unicamp.br : /pub3/linux (Brasil : São Paulo) Slackware
ftp.ime.usp.br : /pub/linux (Brasil : São Paulo) Slackware
ftp.ufpr.br : /pub/Linux/ (Brasil : Paraná) Slackware

Lista de discussões (usernet newsgroup)

comp.os.linux.announce - é um grupo de anúncios moderado; você deve lê-lo se pretende usar Linux. Submissões a este grupo devem ser mandadas para linux-announce@news.ornl.gov.

comp.os.linux.answers - Contém todos os FAQs, HOWTOs e outros documentos importantes. Assine este grupo também.

Os outros grupos na hierarquia comp.os.linux.* também são recomendados algums problemas comuns não respondidos neste FAQ podem estar nos newsgroups. Esses grupos são :

comp.os.linux.setup
comp.os.linux.hardware
comp.os.linux.networking
comp.os.linux.x
comp.os.linux.development.apps,
comp.os.linux.development.system
comp.os.linux.advocacy
comp.os.linux.misc.

Endereço eletrônico de diversos colaboradores da Linux Organização

Drew Eckhard	(US)	drew@coorado .edu
Brondo n S. Allbere	(US)	bsa@kf8nh.wariat.org
Ian Jacksão	(UK)	ijacksão@cus.cam.ac.uk
Rik Faith	(US)	faith@cs.unc.edu
Iar Mc Coghrie	(US)	ian@ucsd.edu
Stephem Harris	(UK)	sweh@spudde .mew.co.uk
DaniOQuilan	(US)	Daniel.Quinlan@linux.org
Fred N. vain Kemper	(US)	waltje@infomagic.com
Mike Sangree	(US)	mike@sojurn.lns.pa.us
Jhon A. Martir	(US)	jmartin@csc.com
David H. Selber	(US)	dhs@gowworm.firefle .com
Chris Netcalf	(US)	metcalf@lcs.mit.edu
Theodo re Tsó	(US)	te tso@athema.mit.edu
Ian Murdo ck	(US)	imurdo ck@debian.org
Stephem Tweedie	(UK)	sct@dcs.ed.ac.uk
David C. Niemi	(US)	niemid@clarck.net